Proc IMechE Part G: Journal of Aerospace Engineering
000(00):1–**

DOI:doi number
http://mms.sagepub.com

# Updating of Aerodynamic Reduced Order Models Generated using Computational Fluids Dynamics*

**L. Griffiths, A. Gaitonde and D. Jones** †
*University of Bristol*

**M. Friswell**
*University of Swansea*

**Abstract**

Reduced order models of CFD codes have been developed to lower computational costs, however each reduced order model has a limited range of validity based on the data used in its construction. Further like the CFD from which it is derived, such models exhibit differences from experimental data due to uncertainty in boundary conditions and numerical accuracy. Model updating provides the opportunity to use small amounts of additional data to modify the behaviour of a reduced order model, which means that the range of validity of the reduced order model can be extended. Whilst here CFD data has been used for updating, the approach offers the possibility that experimental data can be used in future. In this work, the baseline reduced order models are constructed using the Eigensystem Realisation Algorithm and the steps used to update these models are given in detail. The methods developed are then applied to remove the effects of wind tunnel walls and to include viscous effects.

Keywords
Updating, reduced order models, CFD, Euler, RANS

## 1. Introduction

Unsteady aeroelastic simulations require models of both the fluid motion and the structure. CFD methods that solve the full unsteady non-linear Euler or Navier-Stokes equations numerically can give accurate solutions for the fluid, however the high number of degrees of freedom means that they are computationally expensive. This high cost is compounded in the certification of aircraft by the thousands of parameter variations that must be investigated. Therefore historically simpler methods which are not able to predict all the features encountered in the flight regime of modern aircraft have been used e.g. doublet lattice method (DLM) (1, 2). Recent research has focussed on the application of Model Order Reduction

* Electronic supplementary information (ESI) available: Figure data can be found in a data repository (DOI:TBC)
† Corresponding author; e-mail: dorian.jones@bristol.ac.uk

(MOR) schemes to unsteady Computational Fluid Dynamics (CFD) codes (3) as this approach offers a potential increase in accuracy over methods such as DLM. The objective of MOR schemes is to produce a computationally efficient Reduced Order Model (ROM) from the CFD that captures the dominant dynamic behaviour of the full order system. ROMs have been produced for fluid motion using a variety of methods and most often are developed in either the discrete time or discrete frequency domain rather than in continuous time domain, although in some cases a continuous domain ROM can be obtained via transformation.

Reduced order modelling is a term that encompasses a wide variety of methods and there are a number of reviews of the topic including those of Dowell and Hall (3) and Antoulas (4). In this work the ROMs are produced using the Eigensystem Realisation Algorithm (ERA) (5, 6, 7, 8, 9) with restarting (10) that builds on the work of Kung (11). ERA ROMs belong to a category of methods that only focus on a reduced set of known inputs and outputs of the CFD code. For example inputs could be control surface deflections and rates and outputs could be integrated forces rather than flow variables in every cell of the mesh. Only minimal changes are required to the CFD code, with truncated responses to input pulses being sufficient for the MOR process to accurately reproduce the output response of the CFD but at much lower cost (12, 13, 6, 8, 9). It should be noted that where the full order simulation does not accurately reproduce the flow physics, nor will the derived ROM since the ROM can only ever be as accurate as the full order model that has been used to derive it. For this approach the input/output relationship for the fluid flow can be expressed in the continuous time domain as a state space system. Unlike common projection methods, the underlying equations are not reduced and the CFD code does not actually have to be rewritten in state space form. Common limitations of unsteady ROMs (including those generated using ERA) are the assumptions of the CFD used to derive them (for example no viscosity) and the restriction to the calculation of a dynamically linear unsteady flow about a specific non-linear mean flow.

In other fields, such as structural dynamics, tools and methods for model updating have been produced by which models may be made more representative by using experimental data or more accurate simulations. Model updating (14, 15, 16) is a term used in the structural dynamic community to describe a specific type of inverse problem where the goal is to improve the capability of a model to predict a response, as opposed to merely fitting measured data. Model updating requires an initial mathematical model of the physical system, where certain parameters of the system are unknown or uncertain; measured or computational data is then used via a range of methods (14, 17) to estimate these parameters.

In the field of fluid dynamics there has been extensive work on the the correction of DLM using CFD or experimental data (18, 19, 20, 21). However, for many correction techniques used for DLM the correction parameters carry little physical meaning and the process is more analogous to curve fitting. Another example of the use of experimental data to correct a fluids model is the work on Gappy POD (22, 23).

In this paper a new approach to updating state space ROMs is introduced. The method starts from ROMs produced from truncated system responses using the Eigensystem Realisation Algorithm (ERA). Methods to update the steady state and steady state gradients of the resulting flow models are described. Although implemented here using CFD data, the data required is most likely to come from experimental data in practice. Additionally the use of steady response data plus truncated time series is a more efficient approach to CFD based ROMs compared to running long time histories to capture the steady behaviour. Two applications of this method using CFD data are presented as proof of concept for this approach: updating wind tunnel to free air ROMs and inviscid to viscous ROMs.

## 2. Time-continuous Nonlinear and Linear State-space Systems

The basis for the ROMs in this work are the unsteady Euler or RANS equations which are solved using a dual-time version of the standard cell-centred finite-volume scheme code (24, 25) that allows for moving meshes (26, 27, 28). The CFD equations can be written as the following non-linear system:

$$\begin{aligned} \frac{\mathrm{d}\mathbf{x}(t)}{\mathrm{d}t} &= \mathbf{f}(t, \mathbf{x}(t), \mathbf{u}(t)) \\ \hat{\mathbf{y}}(t) &= \mathbf{h}(t, \mathbf{x}(t), \mathbf{u}(t)) \end{aligned} \tag{1}$$

where $(\mathbf{u})$ is the input vector, $(\mathbf{x})$ is a state vector and $(\hat{\mathbf{y}})$ is the output vector, containing changes to quantities of interest. Here the test cases considered are for a 2D aerofoil and the vector $\mathbf{u} = [h, \dot{h}, \alpha, \dot{\alpha}, \delta, \dot{\delta}]^T$, where $h$ is the heave displacement, $\alpha$ is the pitch angle and $\delta$ is the aerofoil flap angle. The vector $\hat{\mathbf{y}}$ contains changes to the unsteady integral forces and moments, but it could contain changes to surface pressures or any other output of interest. The complete unsteady output of the CFD for a motion starting from a non-linear steady state is then the sum of the mean (steady) output $\bar{\mathbf{y}}$ and the dynamic (unsteady) output $\hat{\mathbf{y}}$

$$\tilde{\mathbf{Y}}(t) = \bar{\mathbf{y}} + \hat{\mathbf{y}}(t) \tag{2}$$

Then if the dynamic behaviour of the non-linear Euler or Navier-Stokes system (1) about a non-linear mean or steady flow solution is approximately linear then the non-linear system of equations can be approximated by a linear time-continuous state-space system of the form:

$$\begin{aligned} \frac{\mathrm{d}\mathbf{x}(t)}{\mathrm{d}t} &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) \end{aligned} \tag{3}$$

where the modified dynamic component $\mathbf{y}(t)$ is related to $\hat{\mathbf{y}}(t)$ via $\hat{\mathbf{y}}(t) = \mathbf{y}(t) + \mathbf{D}\mathbf{u}(t)$, where $\mathbf{D}$ is a small known matrix (12).

A modified output of the linearised system is then defined as

$$\mathbf{Y}(t) = \bar{\mathbf{y}} + \mathbf{y}(t) \tag{4}$$

and the output of the non-linear system $\tilde{\mathbf{Y}}(t)$ is approximated by $\mathbf{Y}(t) + \mathbf{D}\mathbf{u}$.

The system is actually solved in discrete time form where the system at time $k\Delta t$ is approximated using a linear backwards difference approximation for the time derivative

$$\begin{aligned} \mathbf{x}_k &= \tilde{\mathbf{A}}\mathbf{x}_{k-1} + \tilde{\mathbf{B}}\mathbf{u}_k \\ \mathbf{y}_k &= \tilde{\mathbf{C}}\mathbf{x}_k \end{aligned} \tag{5}$$

where

$$\begin{aligned} \tilde{\mathbf{A}} &= (\mathbf{I} - \Delta t\mathbf{A})^{-1} \\ \tilde{\mathbf{B}} &= (\mathbf{I} - \Delta t\mathbf{A})^{-1}\mathbf{B}\Delta t \\ \tilde{\mathbf{C}} &= \mathbf{C} \end{aligned} \tag{6}$$

The Eigensystem Realisation Algorithm (ERA) (5) can be used for system identification or model order reduction and requires the discrete Hankel matrices $\tilde{\mathbf{H}}_{ls}(0)$ and $\tilde{\mathbf{H}}_{ls}(1)$ where

$$\tilde{\mathbf{H}}_{ls}(k) = \begin{bmatrix} \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{k}\tilde{\mathbf{B}} & \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{k+1}\tilde{\mathbf{B}} & \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{k+2}\tilde{\mathbf{B}} & \cdots & \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{k+s-1}\tilde{\mathbf{B}} \\ \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{k+1}\tilde{\mathbf{B}} & \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{k+2}\tilde{\mathbf{B}} & \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{k+3}\tilde{\mathbf{B}} & \cdots & \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{k+s}\tilde{\mathbf{B}} \\ \vdots & \vdots & \vdots & & \vdots \\ \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{k+l-1}\tilde{\mathbf{B}} & \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{k+l}\tilde{\mathbf{B}} & \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{k+l+1}\tilde{\mathbf{B}} & \cdots & \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{k+l+s-2}\tilde{\mathbf{B}} \end{bmatrix} \tag{7}$$

where the terms $\tilde{\mathbf{C}}\tilde{\mathbf{A}}^{k}\tilde{\mathbf{B}}$ are the Markov parameters of the system. Initially ERA is used as a model reduction tool for the CFD rather than a system identification tool since the system is of high order, but known. This means that truncated responses can be used and hence that the size of the required Hankel matrices is not too large. In this study a restarted form of the Eigensystem Realisation Algorithm (ERA) (10) has been used as it ensures stable ROMs are produced which is important for an efficient updating process (29). The resulting reduced order discrete system has the following form (5, 10)

$$\begin{aligned} \tilde{\mathbf{x}}_k &= \tilde{\mathbf{A}}_r\tilde{\mathbf{x}}_{k-1} + \tilde{\mathbf{B}}_r\mathbf{u}_k \\ \mathbf{y}_k &= \tilde{\mathbf{C}}_r\tilde{\mathbf{x}}_k \end{aligned} \tag{8}$$

where $\tilde{\mathbf{A}}_r$, $\tilde{\mathbf{B}}_r$ and $\tilde{\mathbf{C}}_r$ are the reduced order system matrices (12).

In the subsequent discussion on updating, this discrete time output consisting of a baseline mean flow $\bar{\mathbf{y}}$ plus the initial ROM for $\mathbf{y}_k$ is referred to as the "original" or uncorrected system. The output of a target system, which a corrected ROM is required to match, can be similarly expressed as:

$$\mathbf{Y}_k^T = \bar{\mathbf{y}}^T + \mathbf{y}_k^T \tag{9}$$

and using a similar notation, the updated ROM then produces a system output which can be written as:

$$\mathbf{Y}_k^{up} = \bar{\mathbf{y}}^{up} + \mathbf{y}_k^{up} \tag{10}$$

To update the unsteady output $\mathbf{Y}_k$ produced by the initial ROM of the uncorrected system, the steady mean components $\bar{\mathbf{y}}$ are updated independently of the unsteady linear terms $\mathbf{y}_k$. Essentially the process constructs a new model of the system with a chosen steady state magnitude and gradient. The description here refers to updating the discrete ROM as this is the approach taken for the results presented. The method to update a continuous time ROM follows an identical process and is described in more detail by Griffiths (29).

## 3. Updating

A correction to the steady state magnitude $\bar{\mathbf{y}}$ is treated as a scalar shift. For any given simulation, the nonlinear mean flow component is overwritten with a known target value from either computational or experimental data.

$$\bar{\mathbf{y}}_{up} = \bar{\mathbf{y}}_T$$

$$\begin{bmatrix} \bar{C}_L \\ \bar{C}_D \\ \bar{C}_M \\ \bar{C}_H \end{bmatrix}_{up} = \begin{bmatrix} \bar{C}_L \\ \bar{C}_D \\ \bar{C}_M \\ \bar{C}_H \end{bmatrix}_T$$

Here $\bar{C}_L, \bar{C}_D, \bar{C}_M$ and $\bar{C}_H$ are the mean lift, drag, moment and hinge moment coefficients, which form the components of the mean output vector $\bar{\mathbf{y}}$.

Updating the reduced order model for the dynamic perturbation to achieve an improved match to the target system can be done in a number of ways. In this study the correction is applied to the input and output matrices as a function of the desired steady state gradient of the system. This approach has been chosen, because this data is readily available from wind tunnel testing. Although it would be simple in this theoretical study to use other data, such data would require significant additional experimental effort to obtain. Here the steady state gradient is taken to be the gradients of the integral forces for a steady state perturbation of heave ($h$), pitch $\alpha$ and flap rotation ($\delta$). The process of updating the unsteady component is equivalent to enforcing the conditions

$$\left.\frac{\partial \mathbf{y}_{up}}{\partial \mathbf{u}}\right|_{SS} = \left.\frac{\partial \mathbf{y}_{T}}{\partial \mathbf{u}}\right|_{SS} \tag{11}$$

where $\left.\frac{\partial \mathbf{y}^{up}}{\partial \mathbf{u}}\right|_{SS}$ and $\left.\frac{\partial \mathbf{y}^{T}}{\partial \mathbf{u}}\right|_{SS}$ are the updated and target steady state gradients of the ROM respectively.

The details of the updating of the ROM are described in the following sections. In Section 3.1, the relationship between steady state gradients and the ROM matrices is described. Sections 3.2 and 3.3 present two alternative methods for correcting the steady state gradients of the ROM so that they match other data. It should be noted here that the original ROM can be created with heavily truncated time series which captures higher frequency terms. This makes it efficient to generate, but may make the prediction of steady state behaviour inaccurate. This inaccuracy is overcome by the updating process described below and is also a quicker approach than converging a time series for longer and not correcting the ROM. Furthermore this steady behaviour need not come from CFD and is exactly the type of data available from experiment.

### *3.1. The state space steady gradients*

The steady state gradients of a discrete ROM are a function of the reduced order system matrices $\tilde{\mathbf{A}}_r$, $\tilde{\mathbf{B}}_r$ and $\tilde{\mathbf{C}}_r$. A new steady state can be found from either a step up or a step down input, see Figure 1.

These discrete step inputs are described by the conditions:

Step down input: Step up input:

$$\begin{aligned} \mathbf{u} &= \begin{cases} \Delta\bar{\mathbf{u}} & \text{for } k <= 0 \\ 0 & \text{for } k > 0 \end{cases} & \mathbf{u} &= \begin{cases} 0 & \text{for } k <= 0 \\ \Delta\bar{\mathbf{u}} & \text{for } k > 0 \end{cases} \\ \mathbf{y} &= \begin{cases} \Delta\bar{\mathbf{y}} & \text{for } k <= 0 \\ 0 & \text{as } k \to \infty \end{cases} & \mathbf{y} &= \begin{cases} 0 & \text{for } k <= 0 \\ \Delta\bar{\mathbf{y}} & \text{as } k \to \infty \end{cases} \end{aligned} \tag{12}$$

Then due to the linearity of the system, the steady state gradient in the limit $k \to \infty$ becomes:

$$\left.\frac{\partial \mathbf{y}}{\partial \mathbf{u}}\right|_{SS} = \frac{\Delta\bar{\mathbf{y}}}{\Delta\bar{\mathbf{u}}} \tag{13}$$

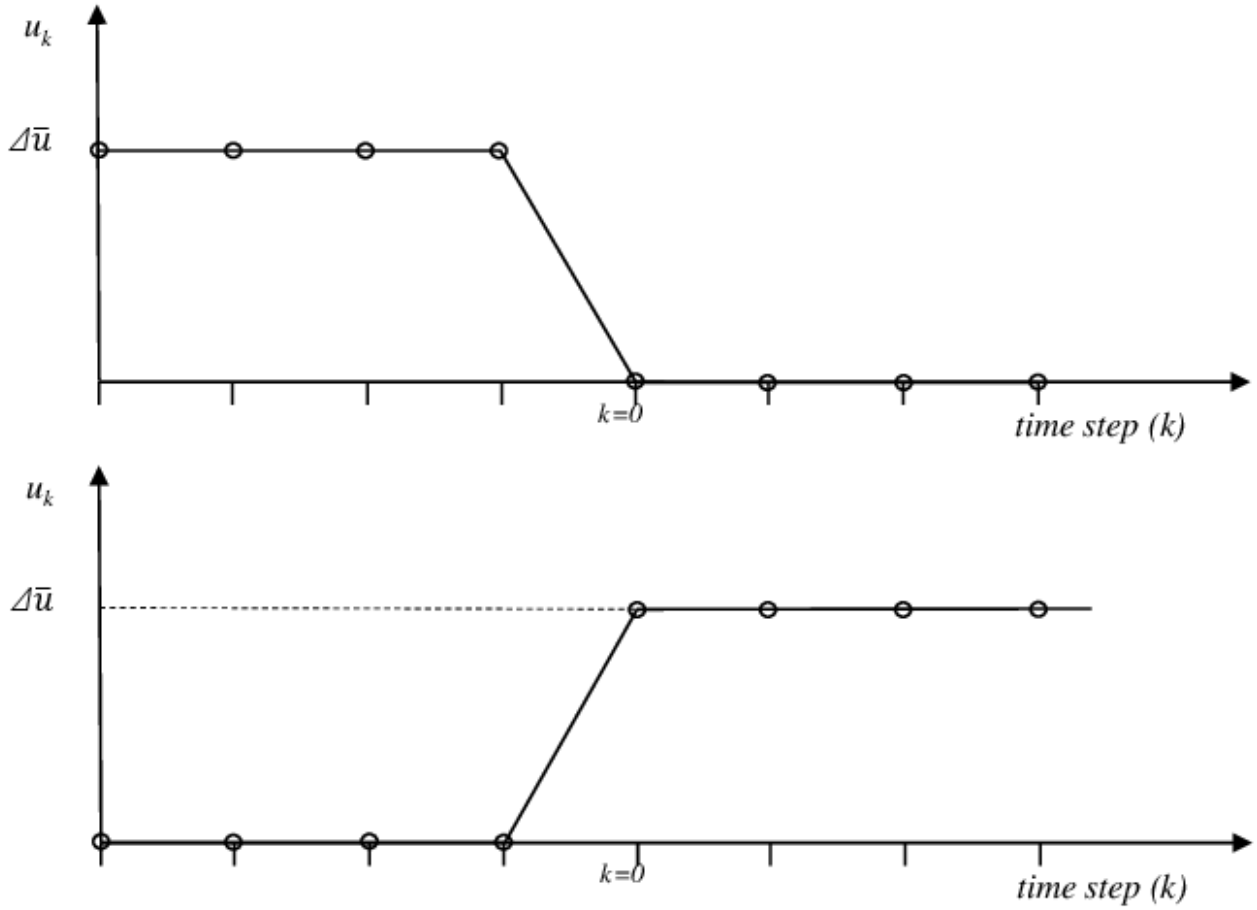


**Figure 1.** Step down pulse input (above) and a step up pulse input (below) for discrete input

Given that the change in input is defined, the steady state response can be found by setting $\mathbf{x}_k = \mathbf{x}_{k-1} = \bar{\mathbf{x}}$ to give

$$(\mathbf{I} - \tilde{\mathbf{A}}_r)\bar{\mathbf{x}} = -\tilde{\mathbf{B}}_r \Delta\bar{\mathbf{u}}$$
$$\Delta\bar{\mathbf{y}} = \tilde{\mathbf{C}}_r \bar{\mathbf{x}}$$

where $\tilde{\mathbf{A}}_r$, $\tilde{\mathbf{B}}_r$ and $\tilde{\mathbf{C}}_r$ are the discrete time reduced order system matrices. Rearranging

$$\frac{\Delta\bar{\mathbf{y}}}{\Delta\bar{\mathbf{u}}} = \tilde{\mathbf{C}}_r \left(\mathbf{I} - \tilde{\mathbf{A}}_r\right)^{-1} \tilde{\mathbf{B}}_r \tag{14}$$

Taking the eigenvalue decomposition of $\tilde{\mathbf{A}}_r$, (14) may be expressed as:

$$\left.\frac{\partial \mathbf{y}}{\partial \mathbf{u}}\right|_{SS} = \tilde{\mathbf{C}}_r \left(\mathbf{I} - \tilde{\mathbf{Q}}\tilde{\mathbf{\Lambda}}\tilde{\mathbf{Q}}^{-1}\right)^{-1} \tilde{\mathbf{B}}_r \tag{15}$$

where $\tilde{\mathbf{\Lambda}}$ is a diagonal matrix formed from the eigenvalues of $\tilde{\mathbf{A}}_r$ and the columns of $\tilde{\mathbf{Q}}$ are the eigenvectors. The eigenvectors $\tilde{\mathbf{Q}}$ can be absorbed into $\tilde{\mathbf{B}}_r$ and $\tilde{\mathbf{C}}_r$ to yield:

$$\left.\frac{\partial \mathbf{y}}{\partial \mathbf{u}}\right|_{SS} = \tilde{\mathbf{C}}_Q (\mathbf{I} - \tilde{\mathbf{\Lambda}})^{-1} \tilde{\mathbf{B}}_Q \tag{16}$$

In more detail, for a model of order $r$ each matrix may be viewed as

$$\tilde{\mathbf{C}}_Q = [\tilde{\mathbf{c}}_1, \tilde{\mathbf{c}}_2, \cdots, \tilde{\mathbf{c}}_n] \qquad \tilde{\mathbf{\Lambda}} = \begin{bmatrix} \tilde{\lambda}_{11} & 0 & \cdots & 0 \\ 0 & \tilde{\lambda}_{22} & & 0 \\ \vdots & & & \vdots \\ 0 & 0 & \cdots & \tilde{\lambda}_{nn} \end{bmatrix} \qquad \tilde{\mathbf{B}}_Q = \begin{bmatrix} \tilde{\mathbf{b}}_1^t \\ \tilde{\mathbf{b}}_2^t \\ \vdots \\ \tilde{\mathbf{b}}_n^t \end{bmatrix} \tag{17}$$

where for a MIMO system $\tilde{\mathbf{b}}_i$ and $\tilde{\mathbf{c}}_i$ are vectors associated with system eigenvalue, $\tilde{\lambda}_{ii}$, whose length is equal to the number of inputs and outputs respectively. Note here that the superscript $t$ denotes the transpose.

### *3.2. Indirect Updating of the Steady State Gradient*

ERA is a moment matching technique where each "moment" is represented as the individual terms of a Hankel matrix. This means the original Hankel matrix of the full order system (7) should be approximated well by the same rank Hankel Matrix produced from the ROM *i.e.*

$$\tilde{\mathbf{H}}_{ls}(k) \approx \begin{bmatrix} \tilde{\mathbf{C}}_r \tilde{\mathbf{A}}_r^k \tilde{\mathbf{B}}_r & \tilde{\mathbf{C}}_r \tilde{\mathbf{A}}_r^{k+1} \tilde{\mathbf{B}}_r & \tilde{\mathbf{C}}_r \tilde{\mathbf{A}}_r^{k+2} \tilde{\mathbf{B}}_r & \cdots & \tilde{\mathbf{C}}_r \tilde{\mathbf{A}}_r^{k+s-1} \tilde{\mathbf{B}}_r \\ \tilde{\mathbf{C}}_r \tilde{\mathbf{A}}_r^{k+1} \tilde{\mathbf{B}}_r & \tilde{\mathbf{C}}_r \tilde{\mathbf{A}}_r^{k+2} \tilde{\mathbf{B}}_r & \tilde{\mathbf{C}}_r \tilde{\mathbf{A}}_r^{k+3} \tilde{\mathbf{B}}_r & \cdots & \tilde{\mathbf{C}}_r \tilde{\mathbf{A}}_r^{k+s} \tilde{\mathbf{B}}_r \\ \vdots & \vdots & \vdots & & \vdots \\ \tilde{\mathbf{C}}_r \tilde{\mathbf{A}}_r^{k+l-1} \tilde{\mathbf{B}}_r & \tilde{\mathbf{C}}_r \tilde{\mathbf{A}}_r^{k+l} \tilde{\mathbf{B}}_r & \tilde{\mathbf{C}}_r \tilde{\mathbf{A}}_r^{k+l+1} \tilde{\mathbf{B}}_r & \cdots & \tilde{\mathbf{C}}_r \tilde{\mathbf{A}}_r^{k+l+s-2} \tilde{\mathbf{B}}_r \end{bmatrix} \tag{18}$$

Updating of the steady state gradient is slightly more complicated for a discrete time ERA ROM compared to a continuous time ERA ROM (29). For a continuous ROM, the continuous Hankel matrices $\mathbf{H}_{ls}(k)$ where $k = 0$ and $k = 1$ are approximated well using the continuous ROM matrices. These have an identical form to the discrete Hankel matrices (18), but with the discrete system matrices replaced by the equivalent continuous system matrices. For the continuous system case, updating can be achieved by approximating $\mathbf{H}_{ls}(k)$ with $k = -1$ and $k = 0$ using the continuous ROM. Then the continuous system steady state gradient appears as the first element of $\mathbf{H}_{ls}(-1)$ and can be replaced by the steady state gradient of the target system. This cannot be done for a discrete ROM since the data required, for negative time, is not available. For discrete systems, a modified process has been developed. Consider a state space system

$$\begin{aligned} \tilde{\mathbf{x}}_k &= \tilde{\mathbf{A}}_r \tilde{\mathbf{x}}_{k-1} + \mathbf{B}_r^* \mathbf{u}_k \\ \mathbf{y}_k &= \tilde{\mathbf{C}}_r \tilde{\mathbf{x}}_k \end{aligned} \tag{19}$$

where $\mathbf{B}_r^*$ is related to the original discrete system matrices ($\tilde{\mathbf{A}}_r$ and $\tilde{\mathbf{B}}_r$) via

$$\mathbf{B}_r^* = (\mathbf{I} - \tilde{\mathbf{A}}_r)^{-1} \tilde{\mathbf{B}}_r \tag{20}$$

Then applying the ERA process to this system (19) will require the Hankel matrices

$$\tilde{\mathbf{H}}_{ls}(0) \approx \begin{bmatrix} \tilde{\mathbf{C}}_r\mathbf{B}_r^* & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r\mathbf{B}_r^* & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^2\mathbf{B}_r^* & \cdots & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^{s-1}\mathbf{B}_r^* \\ \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r\mathbf{B}_r^* & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^2\mathbf{B}_r^* & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^3\mathbf{B}_r^* & \cdots & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^s\mathbf{B}_r^* \\ \vdots & \vdots & \vdots & & \vdots \\ \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^{l-1}\mathbf{B}_r^* & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^l\mathbf{B}_r^* & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^{l+1}\mathbf{B}_r^* & \cdots & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^{l+s-2}\mathbf{B}_r^* \end{bmatrix} \tag{21}$$

and

$$\tilde{\mathbf{H}}_{ls}(1) = \begin{bmatrix} \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r\mathbf{B}_r^* & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^2\mathbf{B}_r^* & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^3\mathbf{B}_r^* & \cdots & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^s\mathbf{B}_r^* \\ \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^2\mathbf{B}_r^* & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^3\mathbf{B}_r^* & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^4\mathbf{B}_r^* & \cdots & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^{s+1}\mathbf{B}_r^* \\ \vdots & \vdots & \vdots & & \vdots \\ \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^l\mathbf{B}_r^* & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^{l+1}\mathbf{B}_r^* & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^{l+2}\mathbf{B}_r^* & \cdots & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^{l+s-1}\mathbf{B}_r^* \end{bmatrix} \tag{22}$$

Then the first term of $\tilde{\mathbf{H}}_{ls}(0)$ is $\tilde{\mathbf{C}}_r\mathbf{B}_r^*$ and

$$\tilde{\mathbf{C}}_r\mathbf{B}_r^* = \tilde{\mathbf{C}}_r(\mathbf{I} - \tilde{\mathbf{A}}_r)^{-1}\tilde{\mathbf{B}}_r \tag{23}$$

which is the steady state gradient of the original ROM (14). The steady gradient term is then replaced in the Hankel matrix to give

$$\tilde{\mathbf{H}}_{ls}(0) \approx \begin{bmatrix} \frac{\partial \mathbf{y}_T}{\partial \mathbf{u}_T}\Big|_{SS} & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r\mathbf{B}_r^* & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^2\mathbf{B}_r^* & \cdots & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^{s-1}\mathbf{B}_r^* \\ \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r\mathbf{B}_r^* & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^2\mathbf{B}_r^* & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^3\mathbf{B}_r^* & \cdots & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^s\mathbf{B}_r^* \\ \vdots & \vdots & \vdots & & \vdots \\ \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^{l-1}\mathbf{B}_r^* & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^l\mathbf{B}_r^* & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^{l+1}\mathbf{B}_r^* & \cdots & \tilde{\mathbf{C}}_r\tilde{\mathbf{A}}_r^{l+s-2}\mathbf{B}_r^* \end{bmatrix} \tag{24}$$

and a new ROM is found using ERA. Initial studies with this indirect approach showed that it worked, however it has two major disadvantages. The first is that a system identification using ERA must be repeated for each new value of the steady gradient. Hence the processes of model construction and updating cannot be separated and any developments become specific to ERA, rather than state space ROMs in general. The second, and more serious disadvantage, is that the ROM attempts to match all the elements of the Hankel matrix in a least squares sense (due to the use of SVD) and so matching the steady gradient cannot be enforced any more than matching any other term of the Hankel Matrix. Hence no results for this method are shown and an alternative direct approach has been developed.

### *3.3. Direct Updating of the Steady State Gradients:*

The method starts by defining an updated steady state gradient as:

$$\frac{\partial \mathbf{y}_{up}}{\partial \mathbf{u}_{up}}\bigg|_{SS} = \frac{\partial \mathbf{y}}{\partial \mathbf{u}}\bigg|_{SS} + \epsilon \tag{25}$$

where each component of the steady state gradient $\frac{\partial \mathbf{y}}{\partial \mathbf{u}}\Big|_{SS}$ (corresponding to each input/output channel of the ROM) is updated individually:

$$\frac{\partial \mathbf{y}_{up}}{\partial \mathbf{u}_{up}}\bigg|_{SS} = \begin{bmatrix} \frac{\partial y_i}{\partial u_j} & \cdots & \frac{\partial y_{imax}}{\partial u_j} \\ \vdots & \ddots & \vdots \\ \frac{\partial y_i}{\partial u_{jmax}} & \cdots & \frac{\partial y_{imax}}{\partial u_{jmax}} \end{bmatrix} + \begin{bmatrix} \epsilon_{i,j} & \cdots & \epsilon_{imax,j} \\ \vdots & \ddots & \vdots \\ \epsilon_{i,jmax} & \cdots & \epsilon_{imax,jmax} \end{bmatrix} \tag{26}$$

and each component of $\epsilon$ is a scalar update parameter. Each update will now be considered separately.

From (25) and (16) it can be seen that a modified steady gradient can be achieved by an update to the input ($\tilde{\mathbf{B}}_{\mathbf{Q}}$) and output ($\tilde{\mathbf{C}}_{\mathbf{Q}}$) matrices of the diagonalised discrete system expressed as:

$$\left.\frac{\partial \mathbf{y}_{up}}{\partial \mathbf{u}_{up}}\right|_{SS} = \left.\frac{\partial \mathbf{y}}{\partial \mathbf{u}}\right|_{SS} + \epsilon = -(\tilde{\mathbf{C}}_Q + \sigma_C \tilde{\mathbf{E}}_{\mathbf{C}}) \left(\mathbf{I} - \tilde{\mathbf{\Lambda}}\right)^{-1} (\tilde{\mathbf{B}}_Q + \sigma_B \tilde{\mathbf{E}}_{\mathbf{B}}) \tag{27}$$

where since each input/output pairing and its update is a SISO system, $\tilde{\mathbf{E}}_{\mathbf{B}}$ and $\tilde{\mathbf{E}}_{\mathbf{C}}$ are $n \times 1$ update matrices calculated *a priori*. The only unknowns are $\sigma_B$ and $\sigma_C$, which are scalar update parameters. As ERA is a balanced system reduction, where the observability and controllability gramians are kept equal in some sense, it was decided to use only a single $\sigma = \sigma_B = \sigma_C$ such that

$$\left.\frac{\partial \mathbf{y}_{up}}{\partial \mathbf{u}_{up}}\right|_{SS} = \left.\frac{\partial \mathbf{y}}{\partial \mathbf{u}}\right|_{SS} + \epsilon = -(\tilde{\mathbf{C}}_Q + \sigma \tilde{\mathbf{E}}_{\mathbf{C}}) \left(\mathbf{I} - \tilde{\mathbf{\Lambda}}\right)^{-1} (\tilde{\mathbf{B}}_Q + \sigma \tilde{\mathbf{E}}_{\mathbf{B}}) \tag{28}$$

where the only unknown is now the scalar parameter $\sigma$.

Rearranging (28)and subtracting (16) yields:

$$\sigma^2 \left[\tilde{\mathbf{E}}_{\mathbf{C}} \left(\mathbf{I} - \tilde{\mathbf{\Lambda}}\right)^{-1} \tilde{\mathbf{E}}_{\mathbf{B}}\right] + \sigma \left[\tilde{\mathbf{C}}_Q \left(\mathbf{I} - \tilde{\mathbf{\Lambda}}\right)^{-1} \tilde{\mathbf{E}}_{\mathbf{B}} + \tilde{\mathbf{E}}_{\mathbf{C}} \left(\mathbf{I} - \tilde{\mathbf{\Lambda}}\right)^{-1} \tilde{\mathbf{B}}_Q\right] + \epsilon = 0 \tag{29}$$

such that the update parameter $\sigma$ is now the solution of a scalar quadratic equation. The quadratic equation (29) has two possible roots for the update parameter $\sigma$. The smallest root is chosen as this introduces the least energy into the system. The discrete updated matrices:

$$\begin{aligned} \tilde{\mathbf{B}}_{up} &= \tilde{\mathbf{B}}_Q + \sigma \tilde{\mathbf{E}}_{\mathbf{B}} \\ \tilde{\mathbf{C}}_{up} &= \tilde{\mathbf{C}}_Q + \sigma \tilde{\mathbf{E}}_{\mathbf{C}} \end{aligned} \tag{30}$$

can then be used as matrices in a new updated ROM, together with the existing $\tilde{\mathbf{A}}_{\mathrm{r}}$ matrix, provided that the eigenvalues are unmodified. If additional eigenvalues are added then the $\tilde{\mathbf{A}}_{\mathrm{r}}$ matrix must also be modified. Both methods are described below.

*Updating using existing eigenmodes*

The update can be isolated onto the input/output of a specific mode of the diagonalised linear system corresponding to the particular eigenvalue. For example the eigenvalue $\tilde{\mathbf{\Lambda}}\Big|_{i,i}$ is isolated by setting both components $\tilde{\mathbf{E}}_{\mathbf{B}}\Big|_i$ and $\tilde{\mathbf{E}}_{\mathbf{C}}\Big|_i$ to be non-zero. Conversely, Setting $\tilde{\mathbf{E}}_{\mathbf{B}}\Big|_i, \tilde{\mathbf{E}}_{\mathbf{C}}\Big|_i = 0$ will mean that the behaviour relating to the eigenvalue $\tilde{\mathbf{\Lambda}}\Big|_{i,i}$ will be unchanged by the updating process. The distribution of $\tilde{\mathbf{E}}_{\mathbf{B}}$ and $\tilde{\mathbf{E}}_{\mathbf{C}}$ has an influence on how heavily the behaviour of a given eigenvalue is modified by the updating. In such a way, the updating process may be isolated to a particular small set of eigenmodes which best characterise the required update to the model.

Eigenvalues that have a large effect on the steady state gradient decay quickly and hence the impact of the update vectors on the pulse response of the system tends to zero very quickly with time so that the updated ROM response tends

back to the original ROM response.

This approach is difficult to implement as defining which mode require modifications is difficult and hence the definition of $\tilde{\mathbf{E}}_{\mathbf{B}}\Big|_i$ and $\tilde{\mathbf{E}}_{\mathbf{C}}\Big|_i$ could be flawed if the right ones are not selected.

*Updating with additional eigenmodes*

An important property of model updating over a simple correction factor is the ability to add extra information to the system i.e. to include a physical response not captured by the original model. For ROM updating this is done by the addition of known eigenvalues and vectors. For example, for the continuous system, to add $j$ additional modes to the diagonalised state space system given by equation (16) ($\tilde{\Lambda}$ , $\tilde{\mathbf{B}}_Q$ , $\tilde{\mathbf{C}}_Q$) with $n$ modes, the updated system becomes:

$$\tilde{\mathbf{A}}_{up} = \begin{bmatrix} \tilde{\mathbf{\Lambda}} & \mathbf{0} \\ \mathbf{0} & \tilde{\mathbf{E}}_{\mathbf{A}} \end{bmatrix} = \begin{bmatrix} \tilde{\Lambda}_1 & & & & \cdots & 0 & 0 \\ & \tilde{\Lambda}_2 & & & & \ddots & 0 \\ & & \ddots & & & & \vdots \\ & & & \tilde{\Lambda}_n & & & \\ \vdots & & & & \tilde{\mathbf{E}}_{\mathbf{A}_1} & & \\ 0 & \ddots & & & & \ddots & \\ 0 & 0 & \cdots & & & & \tilde{\mathbf{E}}_{\mathbf{A}_j} \end{bmatrix} \tag{31}$$

$$\tilde{\mathbf{B}}_{up} = \begin{bmatrix} \tilde{\mathbf{B}}_{\mathbf{Q}} \\ \sigma\tilde{\mathbf{E}}_{\mathbf{B}} \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{B}}_{\mathbf{Q}_1} \\ \tilde{\mathbf{B}}_{\mathbf{Q}_2} \\ \vdots \\ \tilde{\mathbf{B}}_{\mathbf{Q}_n} \\ \sigma\tilde{\mathbf{E}}_{\mathbf{B}_1} \\ \vdots \\ \sigma\tilde{\mathbf{E}}_{\mathbf{B}_j} \end{bmatrix}$$

$$\tilde{\mathbf{C}}_{up} = \begin{bmatrix} \tilde{\mathbf{C}}_{\mathbf{Q}}, \sigma\tilde{\mathbf{E}}_{\mathbf{C}} \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{C}}_{\mathbf{Q}_1} & \tilde{\mathbf{C}}_{\mathbf{Q}_2} & \cdots & \tilde{\mathbf{C}}_{\mathbf{Q}_n} & \sigma\tilde{\mathbf{E}}_{\mathbf{C}_1} & \cdots & \sigma\tilde{\mathbf{E}}_{\mathbf{C}_j} \end{bmatrix}$$

where $\tilde{\mathbf{E}}_{\mathbf{A}_j}$ is the $j^{th}$ additional eigenvalue added to the system. By only adding input/output update components to these additional eigenvalues the additional modes have no influence on the original ROM, and their behaviour is controlled by the solution to (29).

The methods as described thus far, still have a number of implementation issues, which do depend on the application. How should the vectors $\tilde{\mathbf{E}}_{\mathbf{B}_i}$ and $\tilde{\mathbf{E}}_{\mathbf{C}_i}$ or the additional eigenvalues $\tilde{\mathbf{E}}_{\mathbf{A}_i}$ be chosen? This study will consider updating inviscid ROMs from wind tunnel to free air and from inviscid ROMs to viscous ROMs.

## 4. Applications

### *4.1. Inviscid Free Air ROM from a Wind Tunnel ROM*

Consider the case where an accurate discrete-time inviscid model of the flow over a pitching aerofoil in a wind tunnel is available. This consists of a steady state vector and an accurate ROM of the dynamic perturbation. Then the target system is an accurate discrete-time inviscid model of the flow over a pitching aerofoil in free air, consisting of a steady state vector from a full order free air Euler simulation and an accurate ROM of the dynamic perturbation. The question is could the

behaviour of this accurate free air model be reproduced by updating the wind tunnel model using limited data from the free air solution. Note that since the ROMs are based on a dynamically linear assumption only the linear part of the non-linear free air response could be reproduced by the updated model.

First the steady state magnitude of the model outputs are updated via scalar shifts to match the steady state outputs of the target system. The steady state gradients of the original ROM are found algebraically using (16) and are all updated using the direct method described in Section 3.3. For the results presented here the model was also updated with additional eigenvalues. The required update matrices $\tilde{\mathbf{E}}_{\mathbf{A}}$, $\tilde{\mathbf{E}}_{\mathbf{B}}$ and $\tilde{\mathbf{E}}_{\mathbf{C}}$ have to be defined. Here this is done using a "rough" ROM of the target free air system based on truncated responses containing only dominant terms.

Discrete Hankel matrices are then constructed from the accurate wind tunnel ROM and the truncated free air ROM where the discrete time step is identical. The difference between the two Hankel matrices $\Delta\tilde{\mathbf{H}}_{ls}(1)$ is defined by:

$$\Delta\tilde{\mathbf{H}}_{ls}(1) = \tilde{\mathbf{H}}_{ls}(1)\Big|_T - \tilde{\mathbf{H}}_{ls}(1)\Big|_O \tag{32}$$

where the subscript *O* and *T* denote the ROM of the original and target systems. The matrix $\Delta\tilde{\mathbf{H}}_{ls}$ is then used in the ERA method instead of the Hankel matrix itself to identify the update vectors (29) as:

$$\begin{aligned}
\tilde{\mathbf{E}}_{\mathbf{A}} &= {\boldsymbol{\Sigma}_r}^{-1/2}{\mathbf{U}_r}^T\left(\Delta\tilde{\mathbf{H}}_{ls}(1)\right)\mathbf{V}_r{\boldsymbol{\Sigma}_r}^{-1/2} \\
\tilde{\mathbf{E}}_{\mathbf{B}} &= {\boldsymbol{\Sigma}_r}^{1/2}{\mathbf{V}_r}^T\mathbf{E}_m \\
\tilde{\mathbf{E}}_{\mathbf{C}} &= \mathbf{E}_p^T\mathbf{U}_r{\boldsymbol{\Sigma}_r}^{1/2}
\end{aligned} \tag{33}$$

It should be noted that the dynamic update produced from the numerical wind tunnel and free air simulations could also be used to update an equivalent ROM created from experimental, rather than numerical, data. For compressible flows, it is hypothesised that the update is likely to be dominated by inviscid processes and hence using Euler simulations to create the update for an experiment would be sufficient. However viscous updates could also be applied. The Hankel matrix of the target free air Euler system may be of a significantly lower order than the Euler wind tunnel ROM, but as it is only required to capture the dominant modes onto which the updated gradients are applied, it is not necessary to fully capture the low frequency long term response of the target ROM. While the host system typically has a rank of ten to thirty, five to ten modes are sufficient to update the wind tunnel simulation.

In practical application the user must provide a simple model of the desired initial time behaviour, from experiment or computation. The process from that point is then automated so that the eigenvalues updated are based on the size of the singular values of $\Delta\tilde{\mathbf{H}}_{ls}$. In the present application a specified number are included and the effect of varying this number investigated. However it is envisaged that energy considerations such as those used in POD could be used to further automate the process in future.

The test cases in this section simulate a NACA0012 aerofoil undergoing a step motion. The tunnel walls are included as symmetry planes at varying heights. All unsteady simulations start from a mean solution at an incidence of 2.89 degrees and a Mach number of 0.7, where the strong shock in the mean flow means complex interactions occur with the wind tunnel walls when the tunnel height is low.

All the meshes used here for free air and wind tunnel test cases are 251 x 40 with 201 cells over the aerofoil and a first cell height of 0.003c. Examples of the steady meshes used for test cases are shown in Figures 2 and 3. The number

of cells used was identified as being suitable based on an assessment of the effect of doubling and halving the number of cells in each direction on force and moment coefficients, together with run time efficiency, see for example Figure 4. The length of the tunnel in the results was selected based on steady calculations, see for example Figure 5. For $M_\infty = 0.7$ and $\alpha_{mean}$=2.89$^o$ the nonlinear mean flow solutions about which the ROMs are constructed are shown in Figure 6, where it can be seen that the position of the strong shock on the upper surface of the aerofoil is strongly influenced by the position of the wind tunnel walls. When the walls are at 9 chords the shock position is relatively close to the free air position, however by moving the walls much closer to the aerofoil at a height of 2.25 chords the shock wave moves significantly further back on the aerofoil.

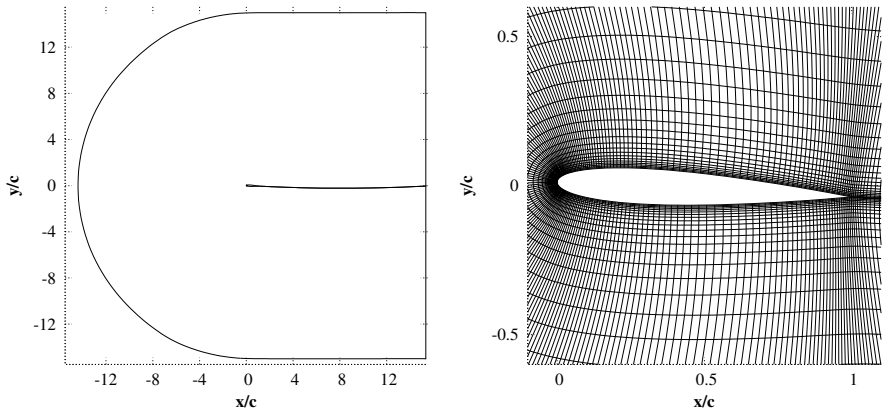


**Figure 2.** Boundary schematic and detail of finite volume Euler freestream mesh for NACA0012 aerofoil.

*Effect of ROM update size on step responses*

The effects of the number of eigenvalues retained for the dynamic update of the steady-state gradients is investigated by considering the response to step inputs in pitch angle $\alpha$ and pitch rate $\dot{\alpha}$. The mesh and initial flow parameters for the step cases are defined in Table 1. Initially accurate ROMs containing 20 modes were constructed from the Euler wind tunnel and free air pulse responses. The Euler wind tunnel ROMs were then updated using differing numbers of eigenmodes. The results are shown in Figures 7 and 8. It can be seen that for both cases the wind tunnel ROM responses exhibit considerably different behaviour from the Euler free air results. An update of any size leads to an improvement in the agreement, however using two eigenmodes is insufficient. Using five eigenmodes gives acceptable levels of agreement, but can lead to the update being weighted solely on the tunnel oscillatory frequencies. The methodology applied here calculates the update parameter $\sigma$ in Equation (29) based on all the eigenmodes describing the update. Large oscillations can occur if the modes captured are predominantly describing the tunnel resonant frequencies. This is avoided by using 10 more eigenmodes. Potential improvements may be possible from excluding the resonant frequencies from the gradient update. Using the direct updating method these modes could be identified from the frequency response of each mode and explicitly excluded, however this has not been implemented.

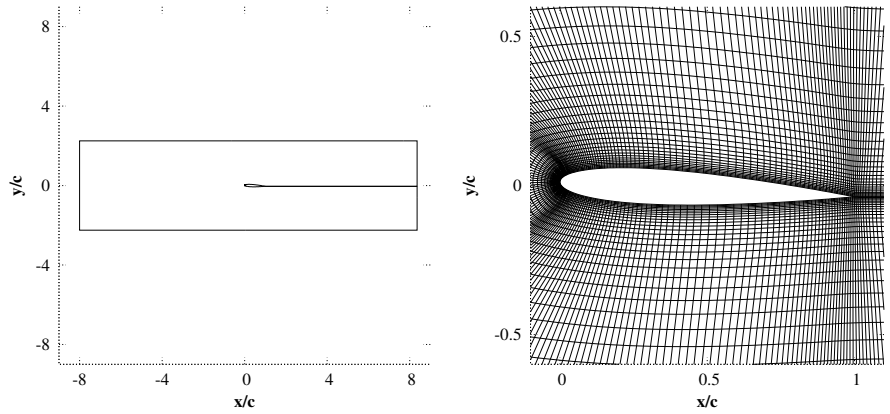


**Figure 3.** Boundary schematic and detail of finite volume Euler tunnel mesh (Height of 4.5 chords) for NACA0012 aerofoil.

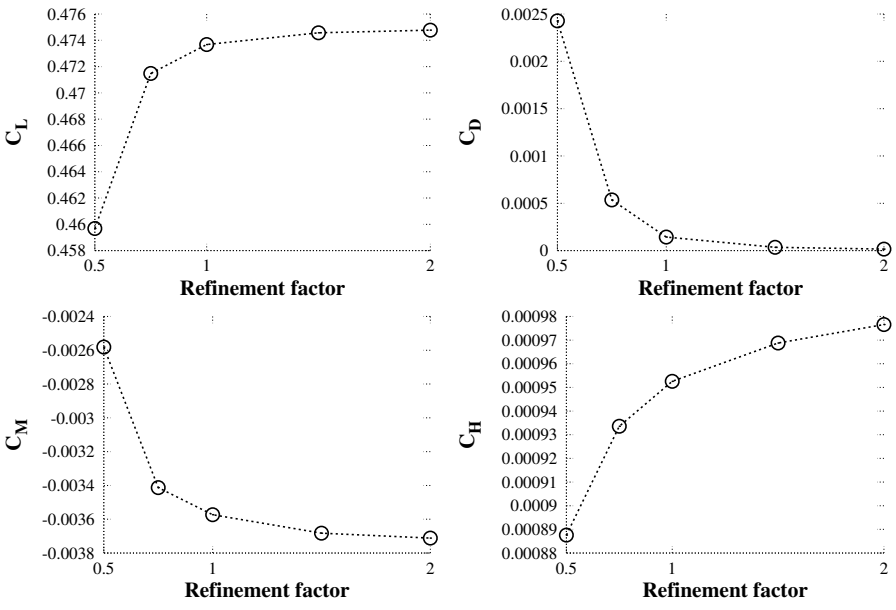


**Figure 4.** Effect of grid density on steady force and moment coefficients.

*Effect of ROM update size on phase angle*

The effects of the number of eigenvalues retained for the dynamic update of the steady-state gradients is further investigated

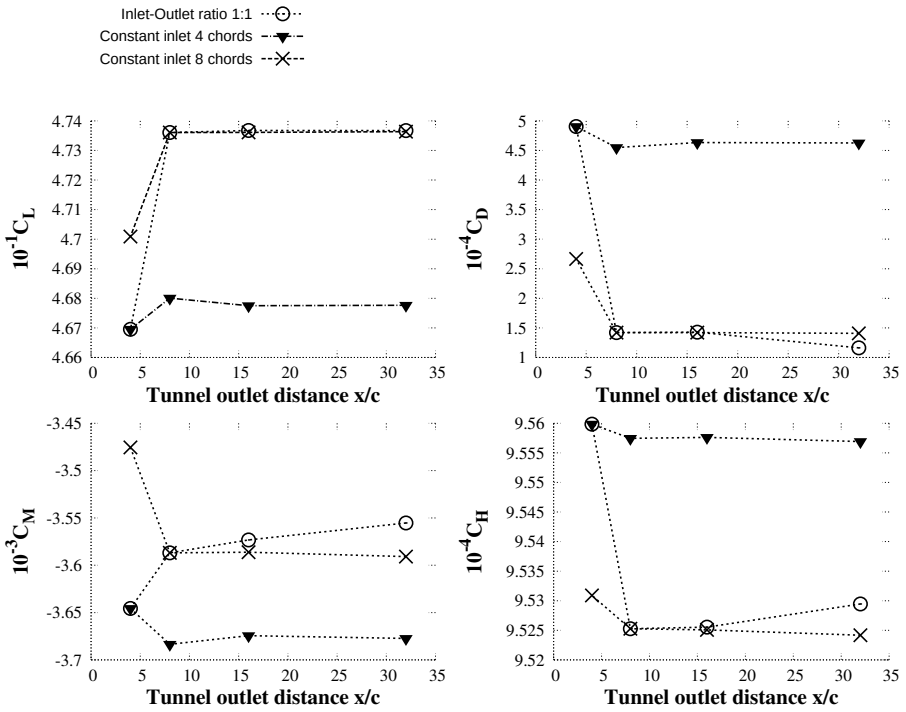


**Figure 5.** Effect of Tunnel Length on Steady Force and Moment Coefficients.

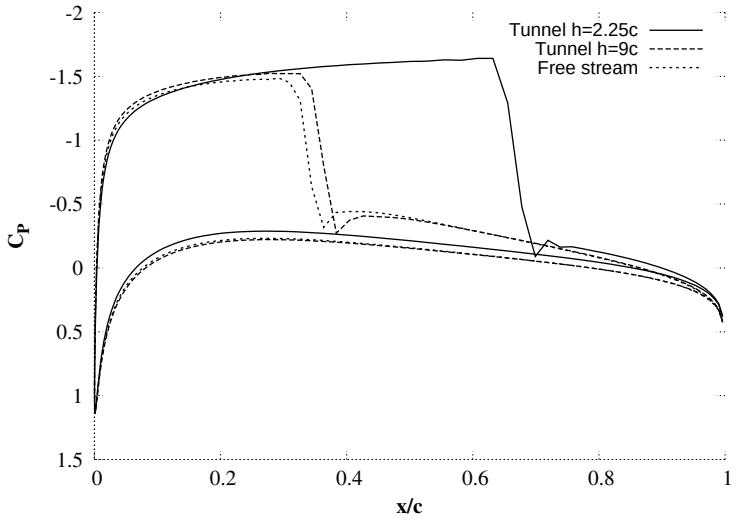


**Figure 6.** Nonlinear mean flow solutions for NACA0012 Aerofoil, $M_\infty = 0.7$, $\alpha_{mean}$=2.89$^o$

| Step Case | $h$ | $M_\infty$ | $\alpha_{mean}$ |
|---|---|---|---|
| 1 | 2.25c | 0.7 | 2.89 |
| 2 | 9c | 0.7 | 2.89 |

**Table 1.** Step test cases

by considering the phase response obtained from the Fast Fourier Transform (30, 31) of impulse responses in pitch angle and pitch rate. The initial solutions are the same as those used for the step responses, see Table 1. Again ROMs containing 20 modes were constructed from the Euler wind tunnel and free air pulse responses. The Euler wind tunnel ROMs were then updated using differing numbers of eigenmodes. The positive frequency results for the test cases are shown in Figures 9 to 10. It can again be seen that for all cases the Euler wind tunnel ROM responses exhibit considerably different behaviour to the Euler free air results. The same conclusions apply: any update gives an improvement, but two eigenmodes is insufficient; using five eigenmodes gives acceptable levels of agreement, but using 10 eigenmodes avoids the update being weighted solely on the tunnel oscillatory frequencies.

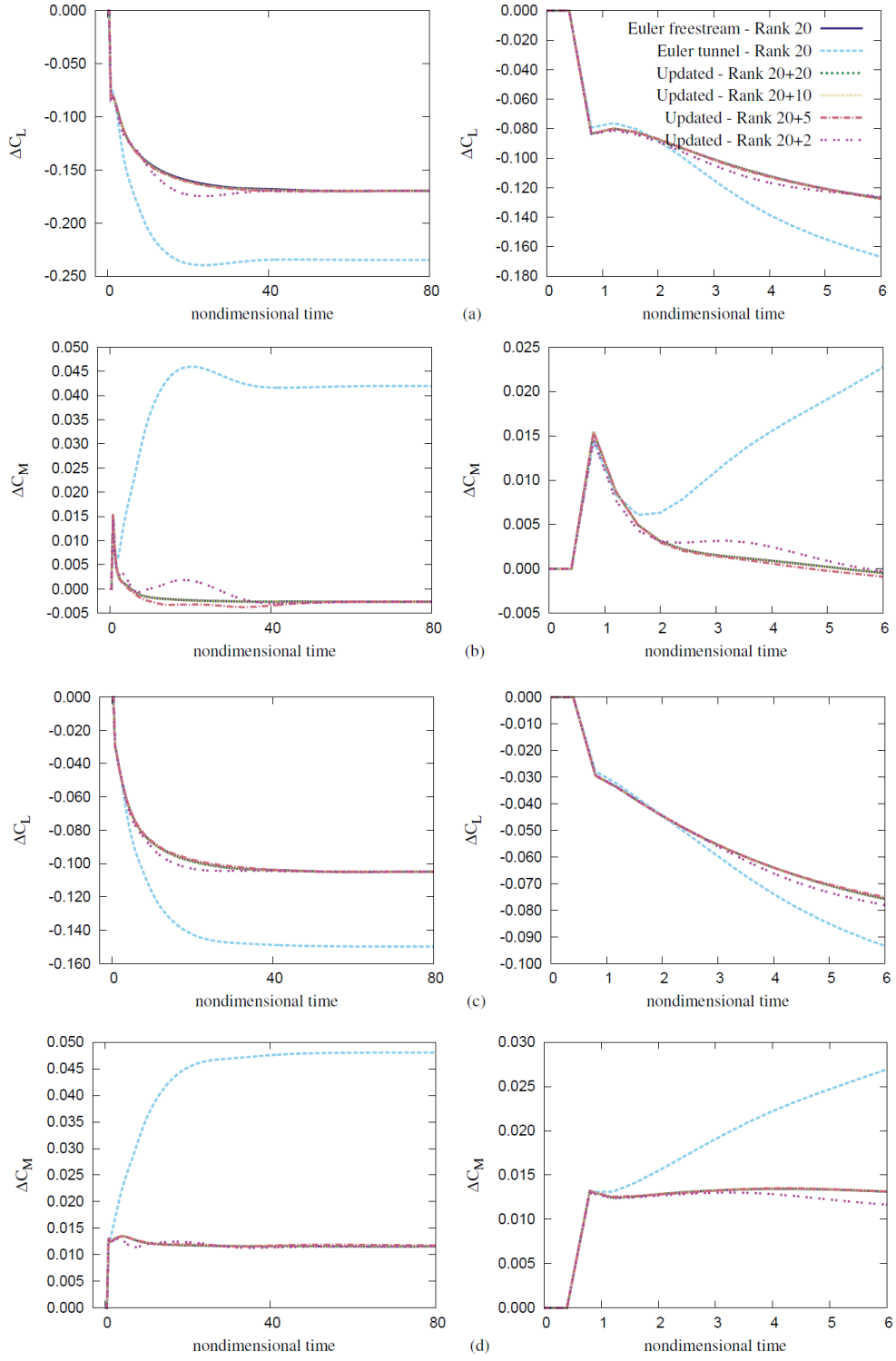


**Figure 7.** Case 1 - Influence of the number of additional eigenvalues on an integral force response to a unit step input. Responses for step in $\alpha$: (a) $\triangle C_L$ (b)$\triangle C_M$ and $\dot{\alpha}$: (c) $\triangle C_L$ (d)$\triangle C_M$

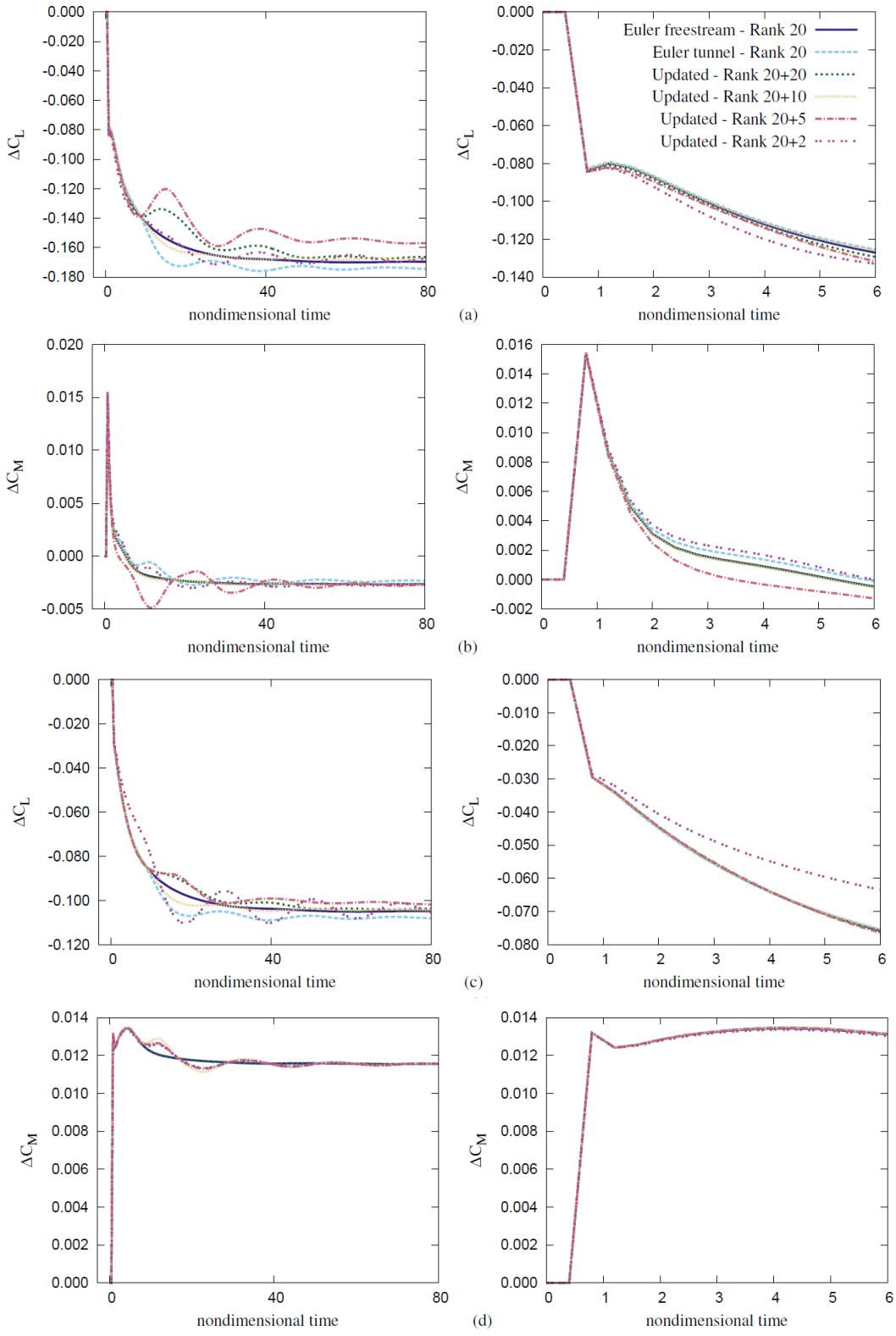


**Figure 8.** Case 2 - Influence of the number of additional eigenvalues on an integral force response to a unit step input. Responses for step in $\alpha$: (a) $\triangle C_L$ (b)$\triangle C_M$ and $\dot{\alpha}$: (c) $\triangle C_L$ (d)$\triangle C_M$

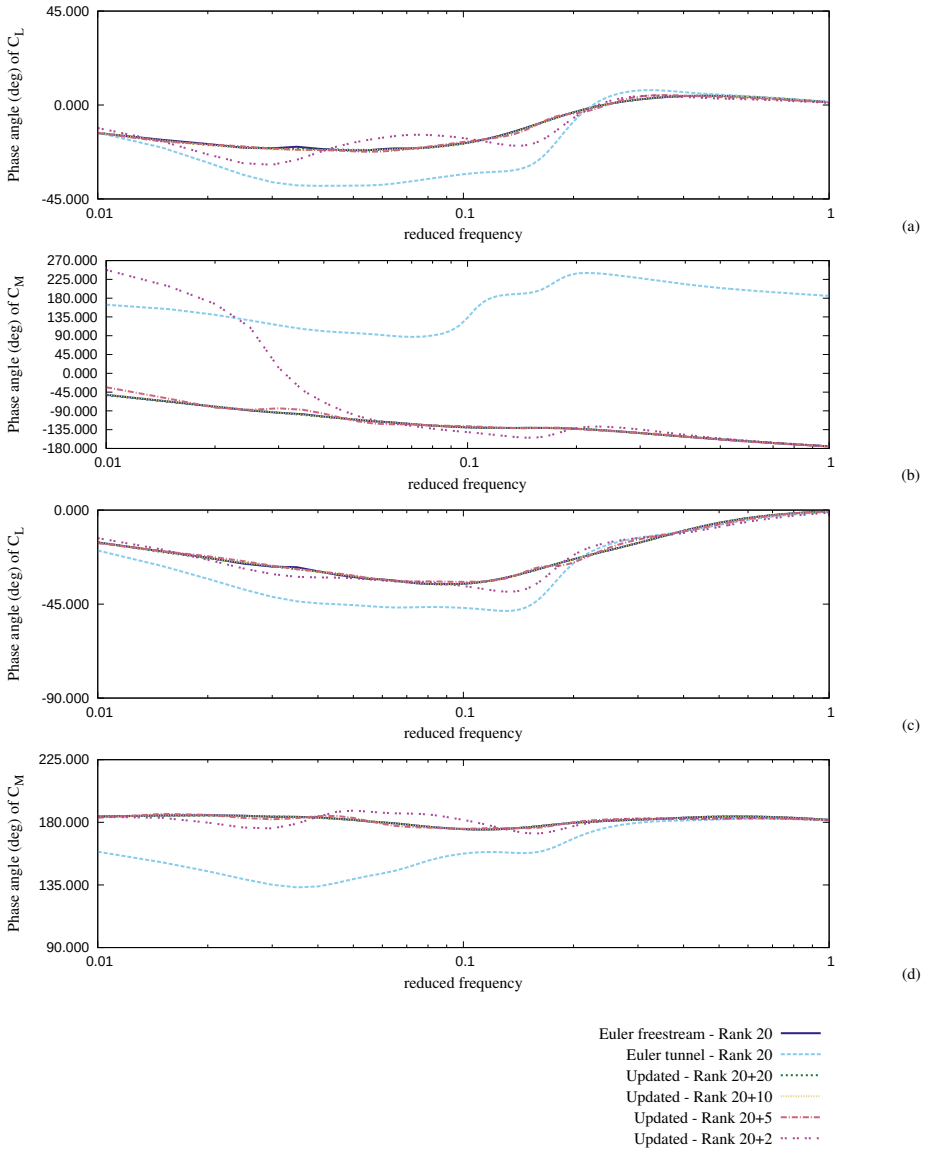


**Figure 9.** Case 1 - Influence of the number of update eigenmodes on the phase angles in the frequency domain. Responses for step in $\alpha$: (a) $\triangle C_L$ (b)$\triangle C_M$ and $\dot{\alpha}$: (c) $\triangle C_L$ (d)$\triangle C_M$

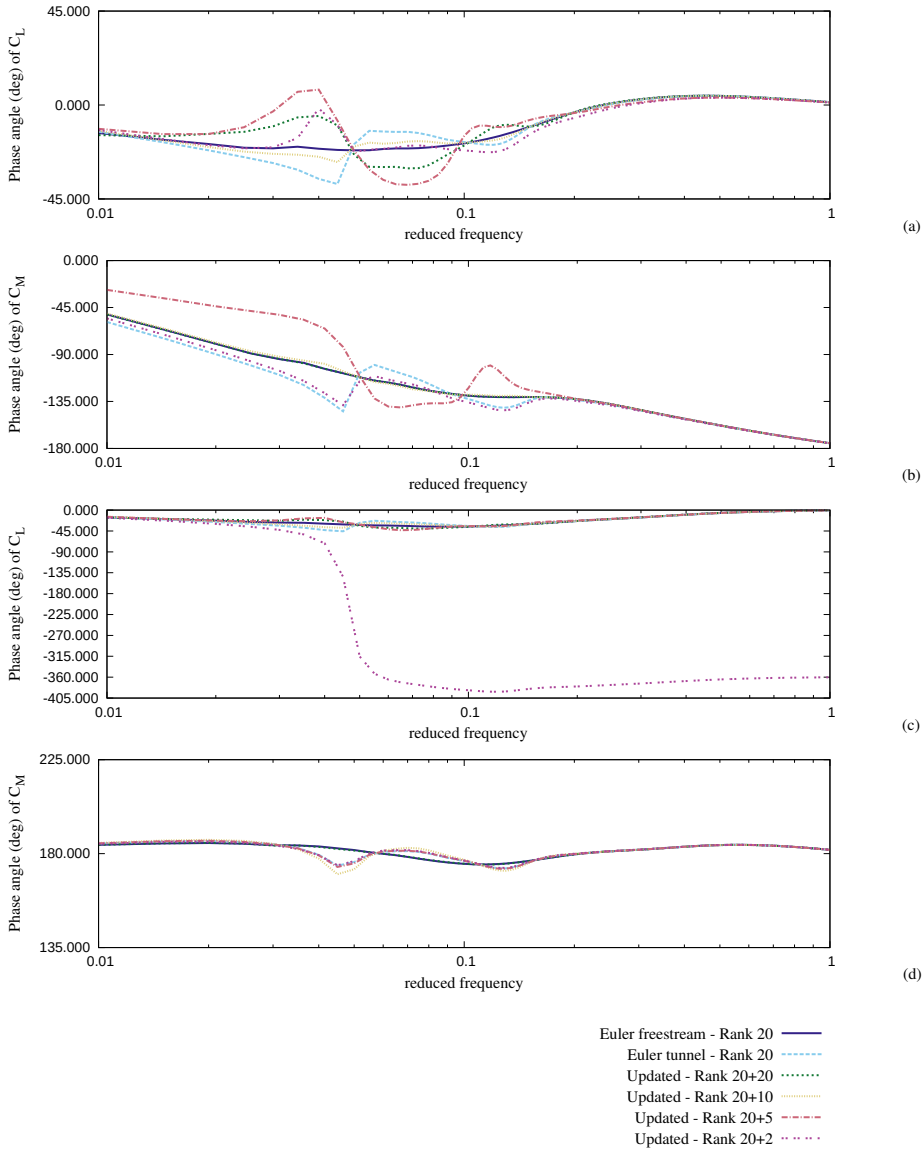


**Figure 10.** Case 2 - Influence of the number of update eigenmodes on the phase angles in the frequency domain. Responses for step in $\alpha$: (a) $\triangle C_L$ (b)$\triangle C_M$ and $\dot{\alpha}$: (c) $\triangle C_L$ (d)$\triangle C_M$

*Effects of wind tunnel wall location and frequency on updating for sinusoidal test cases*

The NACA0012 aerofoil is assumed to be undergoing a linear sinusoidal pitching motion, which is described in dimensional variables by:

$$\alpha(t) = \alpha_{mean} + \alpha_{max} \sin(\omega t^{dim}) \tag{34}$$

where $t^{dim}$ is the dimensional time and $\omega$ is the dimensional frequency, which can be related to the reduced frequency $\kappa$ as:

$$\kappa = \frac{\omega c}{2U_\infty} \tag{35}$$

The parameters for the four test cases are shown in Table 2. It should be noted that although never deployed in the simulations presented in this section, the present code has a region of the trailing edge of the aerofoil defined as a flap with a flap hinge location at 0.75c. The moment of this "flap" about the hinge, $C_H$, is included in the results as it shows that the updating processes work equally well when used locally rather than on global quantities such as lift and pitching moment.

Test case 1 (Figure 11) shows model updating for a small tunnel height of 2.25c. A limitation of the 2D structured mesh generator used resulted in the inlet distance of this mesh being reduced in order to maintain a reasonable cell quality at the leading edge. The resulting inlet length is below the necessary level for a fully mesh independent solution. However this is not important here as even with this mesh it is possible to update it to produce a good quality reduced order model of the free air solution, when compared to the free air solution on a high quality mesh.

The change in shock location from 0.4c to 0.7c in the mean flow, explains the large shifts between the unsteady wind tunnel solution and the unsteady freestream solution. Applying a constant shift to update the steady state magnitude alone is also seen to be insufficient because the angle of the loop relative to the axes is still very different to the freestream loop. The addition of a dynamic update is seen to change the gradient of the loop and the resulting solution is in good agreement with the target free stream solution. Test cases 2-4 show the responses from a more usual tunnel height of 9c at reduced frequencies $\kappa = 0.0404$, $\kappa = 0.0808$ and $\kappa = 0.1616$. Taken as a whole these figures show that as the frequency of oscillation is reduced the steady state update becomes more important, which is not unexpected given that in the limit $\kappa \to 0$ the aerodynamic response tends to the steady state values. Conversely, as the frequency is increased, the impact of the steady state gradients on the frequency response is reduced as the flow becomes increasingly governed by the high frequency eigenmodes of the system (Figures 12-14).

| Case | $\kappa$ | $h$ | $M_\infty$ | $\alpha_{mean}$ | $\Delta\alpha_{max}$ |
|---|---|---|---|---|---|
| 1 | 0.0808 | 2.25c | 0.7 | 2.89 | 1.0 |
| 2 | 0.0404 | 9c | 0.7 | 2.89 | 1.0 |
| 3 | 0.0808 | 9c | 0.7 | 2.89 | 1.0 |
| 4 | 0.1616 | 9c | 0.7 | 2.89 | 1.0 |

**Table 2.** Frequency test cases

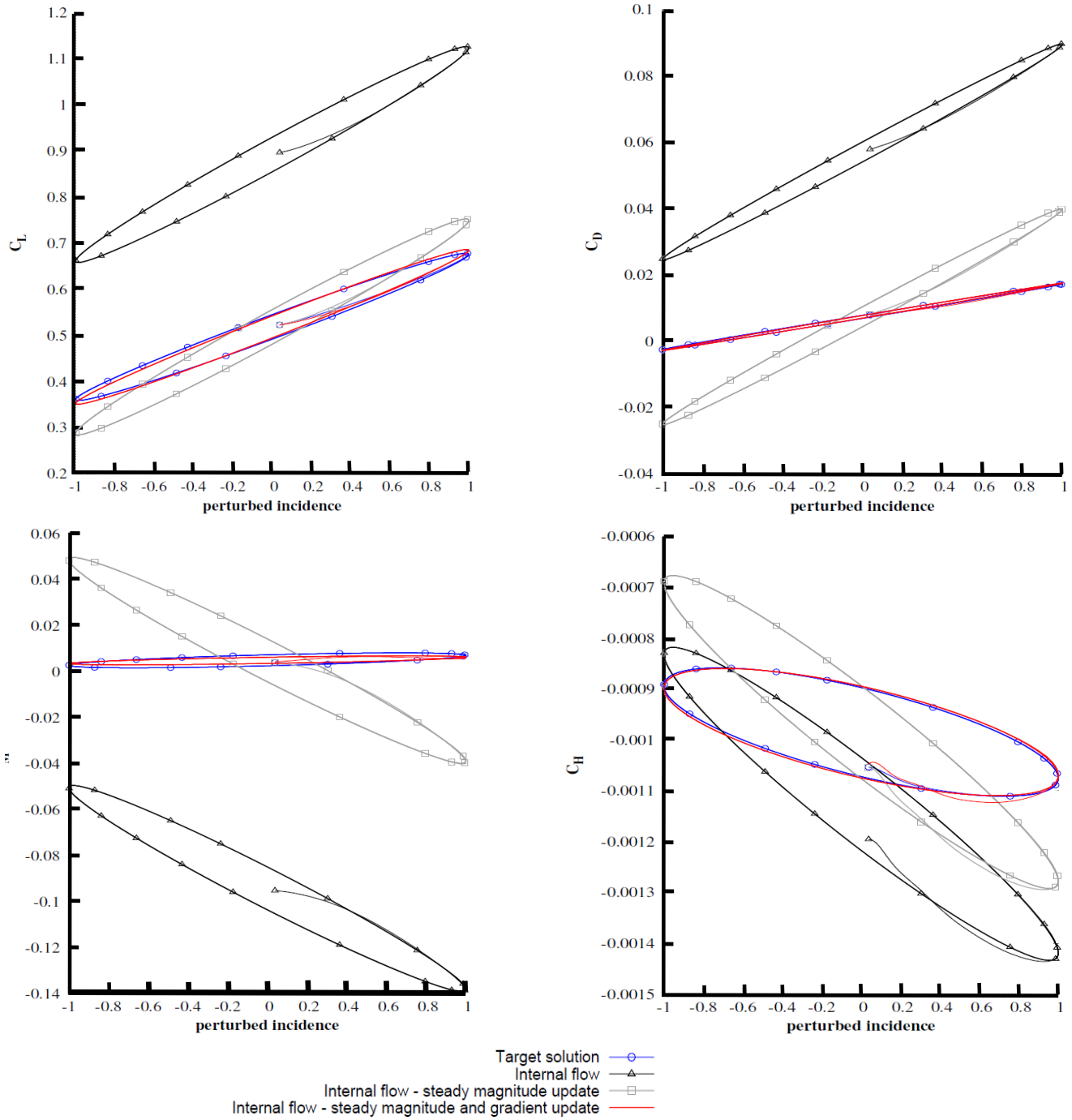


**Figure 11.** Test case 1- Integral force response ($\kappa = 0.0808$)

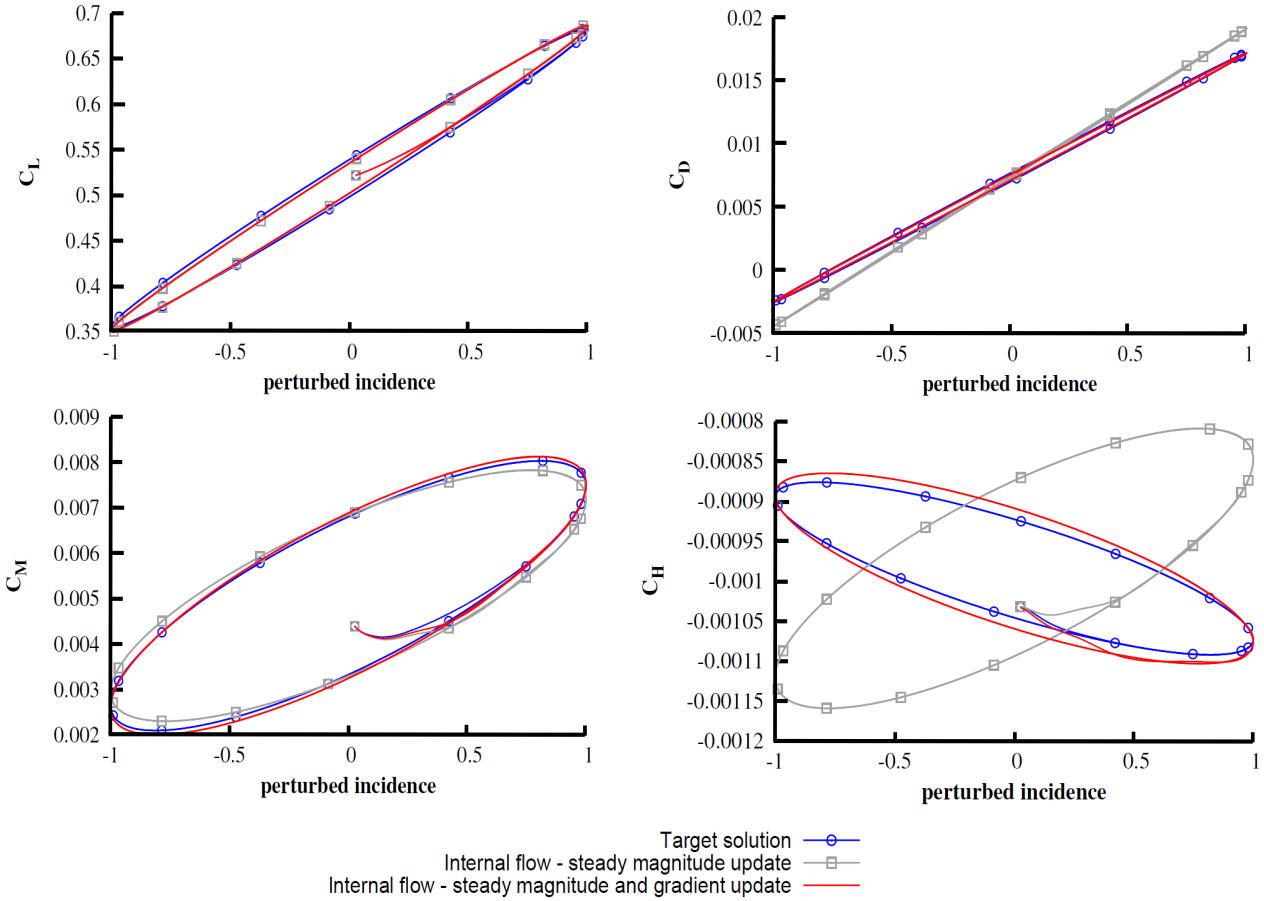


**Figure 12.** Test Case 2 - Integral force response ($\kappa = 0.0404$)

*Practical Application to Experimental Wind Tunnel Data*

The computational procedure outlined above can be used to determine corrections to both the steady state and the unsteady perturbation. The computationally obtained correction to the unsteady part could be used to update an experimentally derived model based on the actual wind tunnel data, thus giving improved modelling of unsteady free air flow. The accuracy of this update procedure will depend on the importance of the viscous effects, since the updates will have come from an inviscid model. The above process could be repeated with a RANS solver to overcome this difficulty. An alternative would be to additionally update the ROM for the effects of viscosity. To illustrate that the updating approach described in this paper can be applied to other problems, some initial results from an investigation of updating an inviscid ROM to include viscous effects is presented in the next section.

## *4.2. A Viscous ROM from a Inviscid ROM*

Essentially the updated ROMs described in this paper are all dynamically linear and the question arises as to whether such ROMs have any value for viscous flows. It has been noted in a number of studies that turbulence models coupled with the RANS equations do not necessarily interact with the frequency and amplitude of a forced unsteady motion (32). Indeed the concept of a mean turbulent boundary layer has also been shown experimentally: Brereton *et al* (33) tested freestream unsteadiness in the frequency range 0.1-2.0 Hz and concluded that even when the unsteadiness departs substantially from

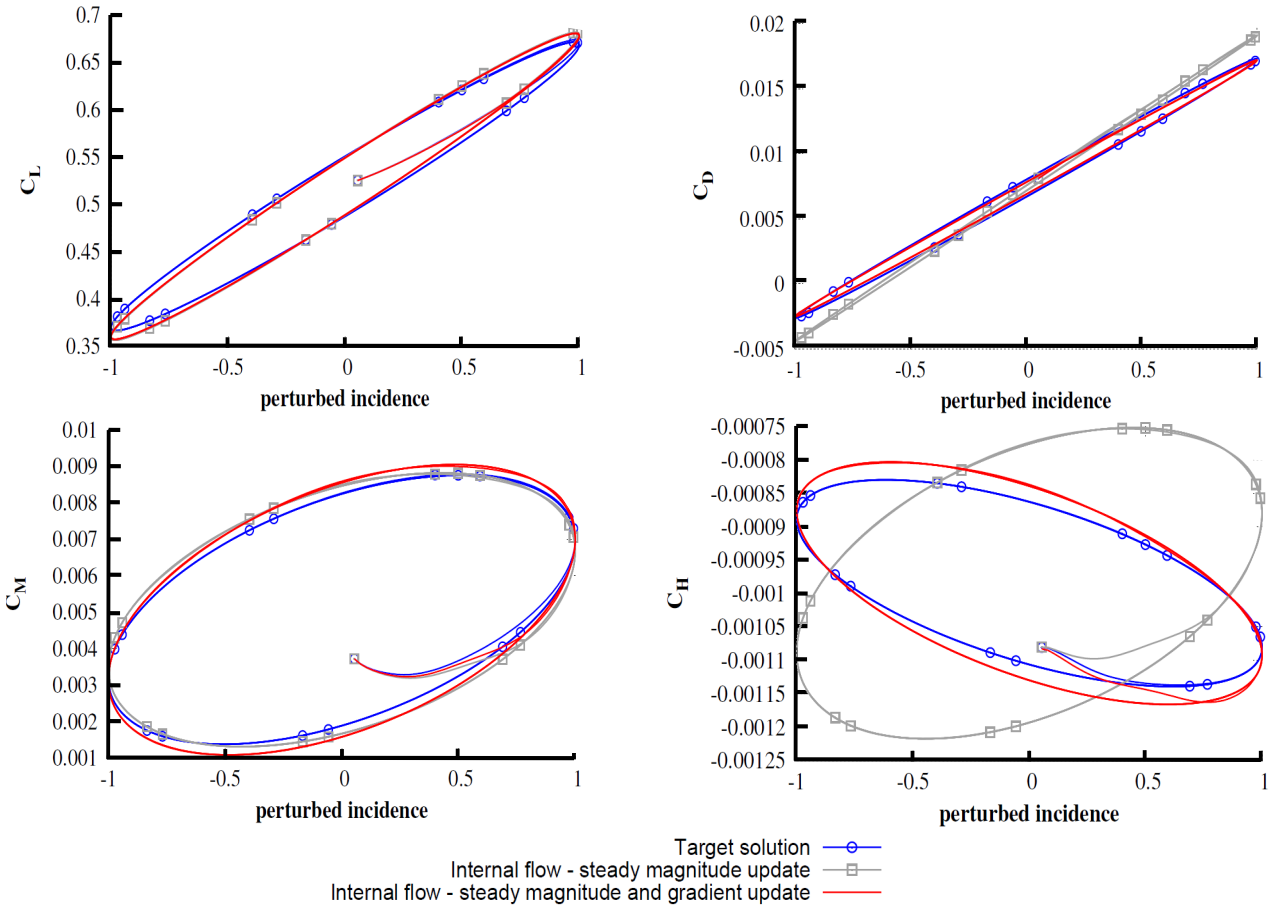


**Figure 13.** Test Case 3 - Integral force response ($\kappa = 0.0808$)

its quasi steady state the unsteady boundary layer remained robust to its steady state condition. Hence the assumption of dynamic linearity of the RANS equations that are implicit in the updated model produced by the current process (29) is reasonable (34, 35) for many situations.

The test cases in this section simulate flow over either a NACA0012 and a RAE2822 aerofoil pitching about its quarter chord. The results for the RAE2822 are at a Mach number of 0.8 where both the Euler and RANS solutions exhibit strongly shocked solutions in the mean flow. All results for the NACA0012 are given at a Mach number of 0.6, which corresponds to the Mach number of the AGARD test case CT1, where the mean flow is shock free.For the NACA00012 test cases, the size of the mean free air Euler mesh is $281 \times 40$ with 201 points on the aerofoil and the RANS mesh is $401 \times 50$ with 301 points on the aerofoil and a $y^+$of approximately 1.0 for the undisturbed mesh. For the RAE2822 test the mean free air Euler mesh is coarse with size $139 \times 15$ and 99 points on the aerofoil. The RANS mesh has the same size and $y^+$ as for the NACA0012.

The initial results presented here start from an inviscid flow model, consisting of an inviscid non-linear steady state vector and an accurate linear inviscid ROM of the dynamic perturbation. This is updated to produce a model of the target system that is a viscous RANS model (29). The non-linear steady state magnitude is first updated to match the non-linear viscous mean flow of the target system. The dynamic ROM must then be updated. The update vectors are produced using a "rough"

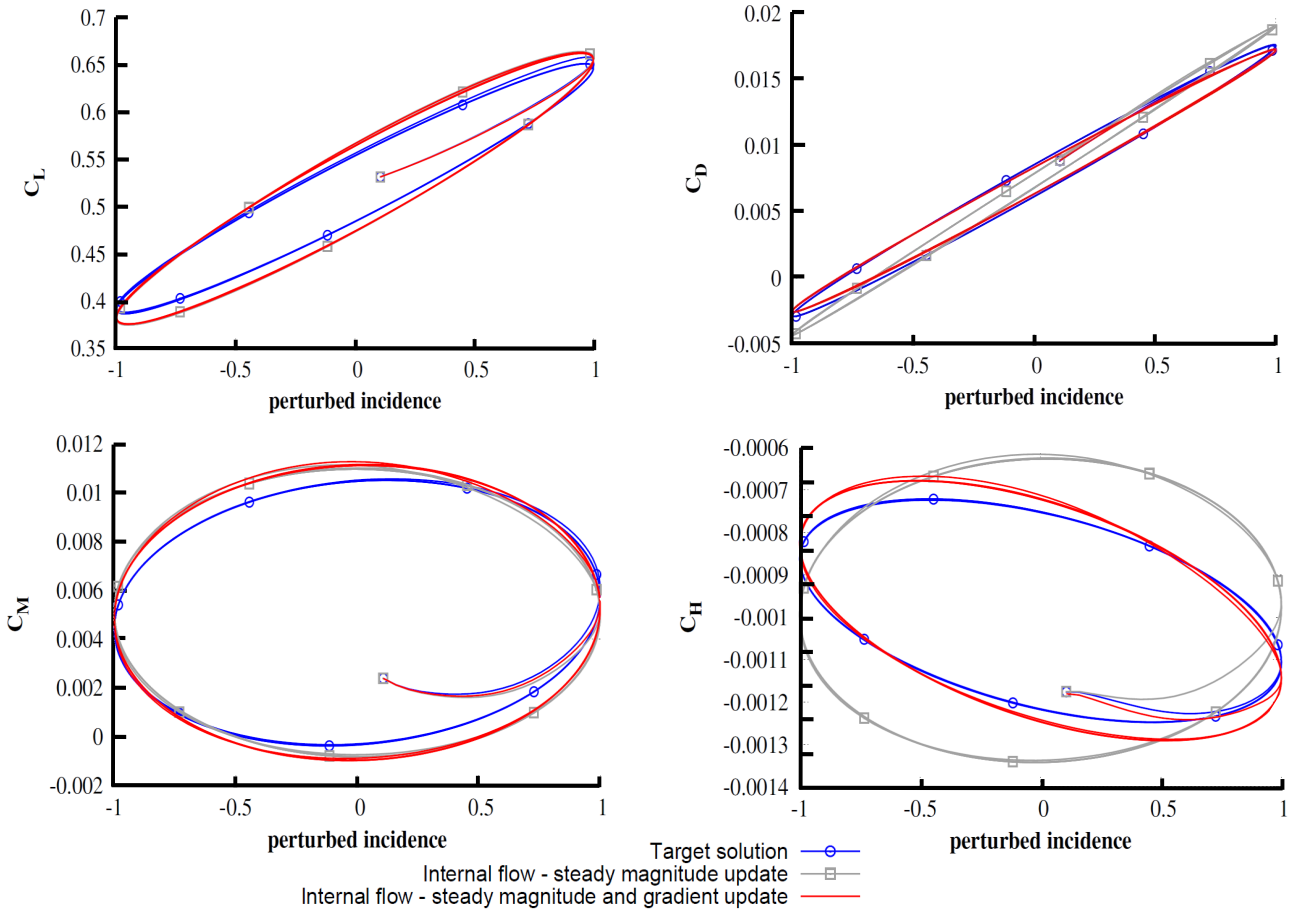


**Figure 14.** Test Case 4 - Integral force response ($\kappa = 0.1616$)

or inexact ROM of the target system produced using heavily truncated pulse responses to minimise computational costs and the steady-state gradients which may be obtained from the difference of two steady-state solutions of the target system as described in detail by Griffiths (29). In a practical application this data for updating would come from experimental data. However even as a method for producing a computational RANS ROM this approach is more computationally efficient than producing a RANS ROM directly. This is because only very truncated RANS simulations are required coupled with an Euler ROM which has been developed from simulations on much coarser meshes. To directly produce a RANS ROM requires far longer time responses on much finer meshes than are necessary with the current approach otherwise steady state behaviour would not be correctly predicted by the ROM.

*Effect of ROM update size on step responses*

The effects of the number of eigenvalues retained for the dynamic update is investigated by considering the response to step inputs in pitch angle $\alpha$ and pitch rate $\dot{\alpha}$. Initially accurate ROMs containing 20 modes were constructed from the Euler and RANS free air pulse responses for the two cases shown in Table 3. Since the updated ROMs can only match at best reproduce the dynamically linear component of the RANS response this accurate dynamically linear ROM of the RANS is used for comparison. The Euler ROMs were then updated using differing numbers of eigenmodes obtained from a lower fidelity inexact RANS model, constructed from 80 Markov parameters (more detail is given in Griffiths (29)). The results for these step test cases are shown in Figures 15 and 16. It can be seen that for all cases the responses predicted by the Euler

| Step Case | Aerofoil | $Re$ | $M_\infty$ | $\alpha_{mean}$ |
|---|---|---|---|---|
| 1 | NACA0012 | $4.8 \times 10^6$ | 0.6 | $2.89^o$ |
| 2 | RAE2822 | $4.8 \times 10^6$ | 0.8 | $0^o$ |

**Table 3.** Step test cases

ROM free air results exhibit considerable differences in behaviour compared to the RANS free air results. An update of any size leads to an improvement in the agreement, however using five or more eigenmodes offered a considerable increase in accuracy over only two modes.

*Effect of ROM update size on phase angle*

The effects of the number of eigenvalues retained for the dynamic update is further investigated by considering the phase response obtained from the Fast Fourier Transform (31, 30) of impulse responses in pitch angle and pitch rate. The positive frequency results are shown in Figure 17 and 18. It can again be seen that for all cases the Euler ROM responses exhibit different behaviour to the RANS free air ROM results. An update of any size leads to an improvement in the agreement, however using two eigenmodes is insufficient. Using five eigenmodes gives acceptable levels of agreement.

Further results showing the effect of using different numbers of Markov parameters in the inexact ROM are available in Griffiths (29).

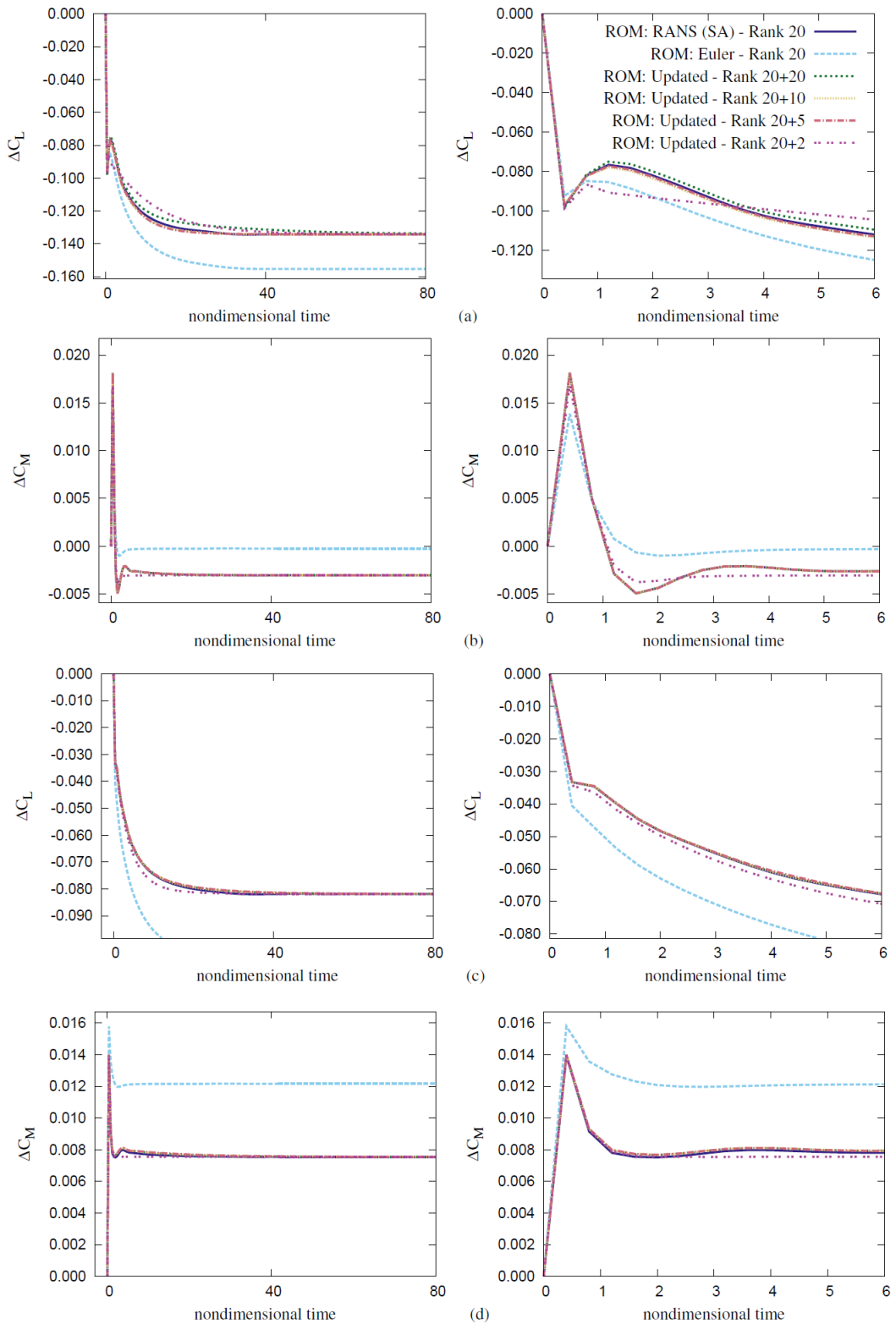


**Figure 15.** Case 1 - Influence of the number of additional eigenvalues on an integral force response to a unit step input. Responses for step in $\alpha$: (a) $\triangle C_L$ (b) $\triangle C_M$ and $\dot{\alpha}$ (c) $\triangle C_L$ (d) $\triangle C_M$

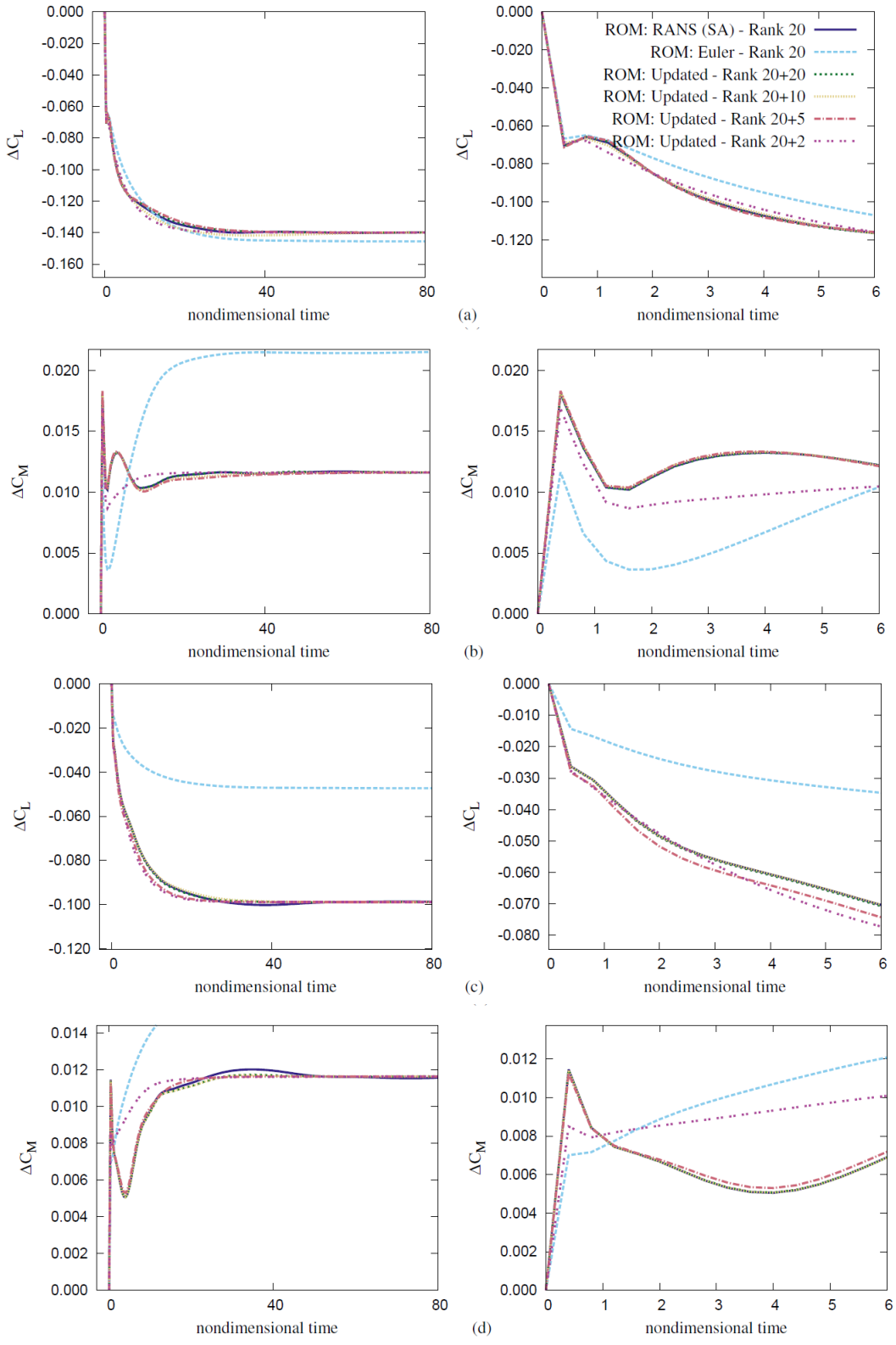


**Figure 16.** Case 2 - Influence of the number of additional eigenvalues on an integral force response to a unit step input. Responses for step in $\alpha$: (a) $\triangle C_L$ (b) $\triangle C_M$ and $\dot{\alpha}$ (c) $\triangle C_L$ (d) $\triangle C_M$

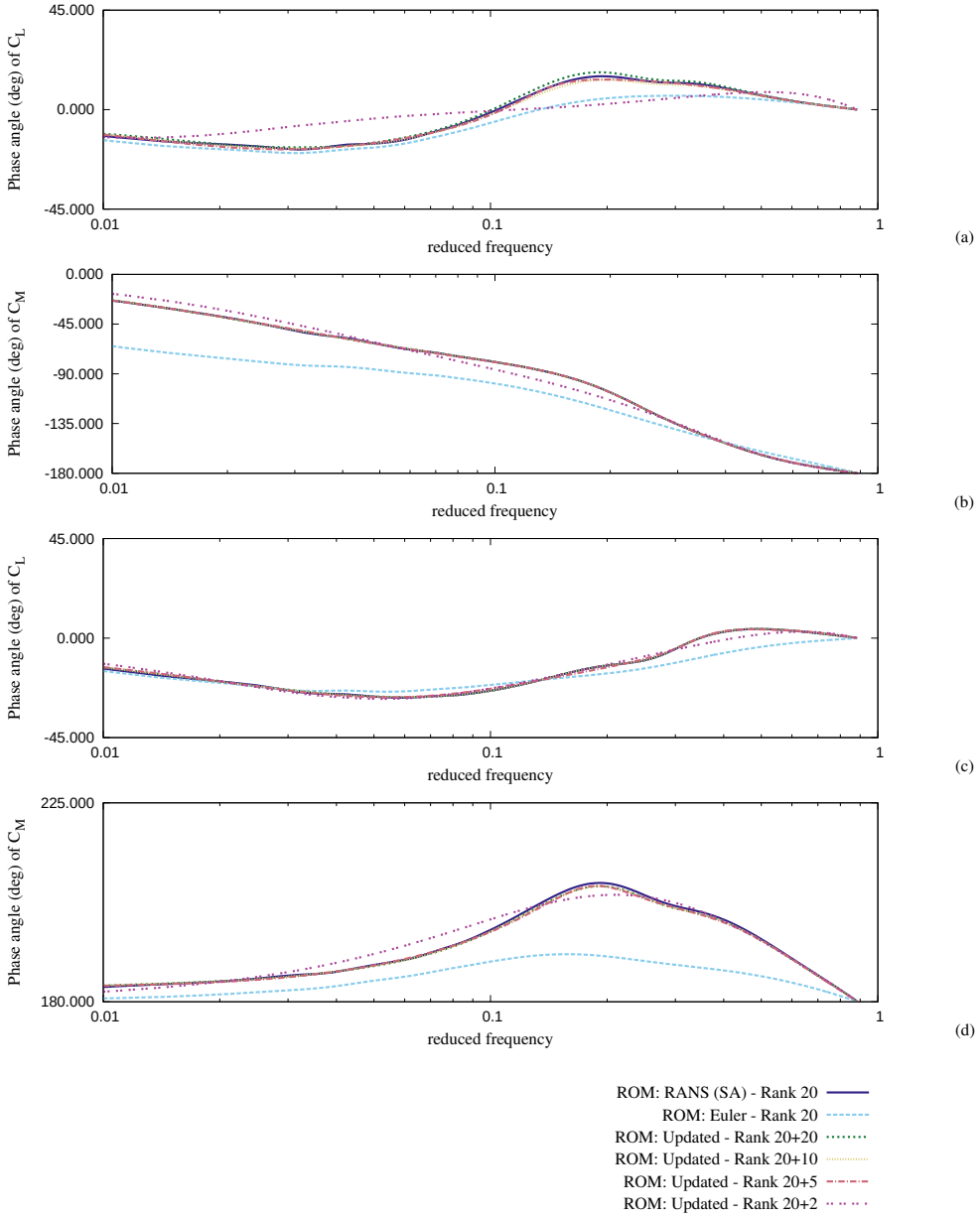


**Figure 17.** Case 1 - Influence of the number of additional eigenvalues on the phase angles in the frequency domain. Responses for impulses in $\alpha$: (a) $\triangle C_L$ (b) $\triangle C_M$ and $\dot{\alpha}$ (c) $\triangle C_L$ (d) $\triangle C_M$

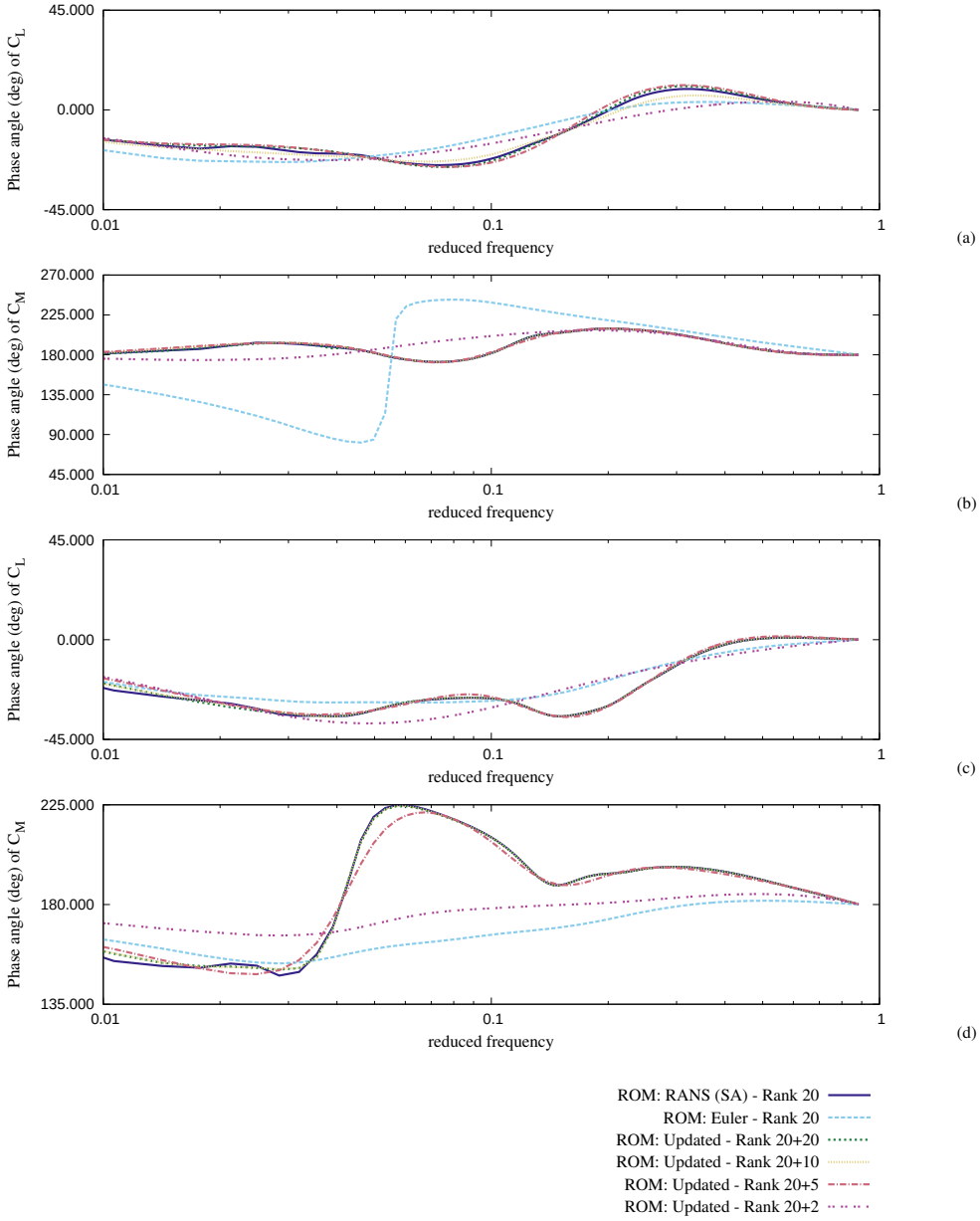


**Figure 18.** Case 2 - Influence of the number of additional eigenvalues on the phase angles in the frequency domain. Responses for impulses in $\alpha$: (a) $\triangle C_L$ (b) $\triangle C_M$ and $\dot{\alpha}$ (c) $\triangle C_L$ (d) $\triangle C_M$

*AGARD CT1*

The final case shown here is the AGARD test case CT1 (36, 37). In each of the figures shown there are a number of solutions in addition to the experimental data. These are summarised in Table 4

| Label | Description |
|---|---|
| Nonlinear freestream RANS | Standard non-linear RANS simulation of the unsteady flow |
| ROM: RANS (target) | This is a viscous ROM created directly from the nonlinear RANS solution using 400 Markov parameters |
| ROM: Euler (original) | This is an Euler ROM for the CT1 test case calculated on a coarser mesh |
| ROM: Updated 10 M.P. rank 20+5 | This is Euler ROM updated using the process described in Section 4, with 20 Markov parameters |

**Table 4.** Step test cases

The integral force responses to the sinusoidal pitch input are shown in Figure 19 where the Euler ROM is updated using an inexact RANS ROM with 10 modes constructed using just 20 Markov parameters i.e. from twenty time steps. For each update the maximum number of modes which can be identified with ERA is half the number of Markov parameters. The results show good correlation with the accurate RANS ROM constructed using 400 Markov parameters. Note that the accurate linear ROM of the viscous RANS is the target that the dynamically linear updated ROM is attempting to match. Since the ROM is linear it cannot reproduce dynamically nonlinear behaviour.

In Figure 19 it can be seen that at the extremes of the motion, all the linear models are no longer in agreement with the nonlinear solution. The point around which the linear models have been constructed does not contain a shock nor trailing edge separation, which are present in the non-linear solution. Hence this invalidates the approximation of a dynamically linear perturbation about a non-linear mean. The experimental data for lift and moment are not in agreement with the non-linear RANS CFD especially for the moment coefficient. Here it can be seen that an updating using experimental data would be required to update the inviscid ROM closer to these conditions. Ongoing research will address this and additionally methods for bringing nonlinear effects into these linear ROMs is in progress (38).

## 5. Conclusions

A new approach to updating state space ROMs has been described in this paper. In practice it is anticipated that the approach will combine small amounts of experimental data with an initial Euler ROM of the system. However, in this proof of concept study the additional data is provided by CFD rather than experiment. This approach also has a further benefit for purely computational ROMs of large systems as it allows ROMs to be created from short truncated responses, more cheaply than obtaining ROMs of similar accuracy from extended responses. The original ROMs have been produced using the Eigensystem Realisation Algorithm (ERA)and the steady state and steady state gradients have then been updated using a direct updating approach, which works better than the indirect approach that is also described. The initial results presented here show that model updating of ROMs has great potential, with ROMs which has been developed for a different situation being corrected to ROMs that work for a new situation. It should be noted that as linear ROMs, even the corrected ROMs will be not be able to model non-linear behaviour though methods to include non-linearity in the linear ROM are being investigated at Bristol. Further work is still needed before model updating of ROMs can be applied between any flow configurations and use experimental data.

## Acknowledgements

The authors would like to acknowledge EPSRC, who funded this work.

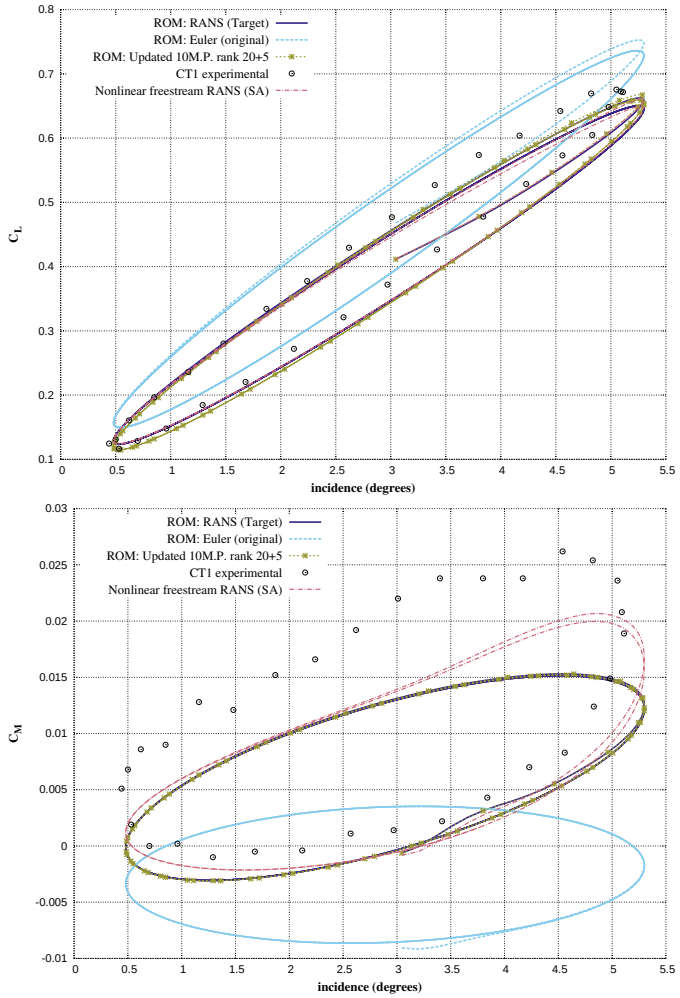


**Figure 19.** Unsteady motion of the AGARD CT1 test case in free air. Updated with an inexact ROM with 20 Markov parameters and 10 modes.

## Data Access

The data used in the figures presented in this paper are available at DOI:TBC

### References

[1] Albano, E. and Rodden, W.P. (1969) A Doublet Lattice Method for Calculating Lift Distributions on Oscillating Surfaces in Subsonic Flows, AIAA Journal, Vol. 7, No. 2, p 279-285.

[2] Blair, M. (1992) A compilation of the mathematics leading to the doublet lattice method, U.S. Air Force Wright Laboratory, WL-TR-92-3028.

[3] Dowell, E.H. and Hall, K.C.(2001) Modelling of fluid-structure interaction, *Annual Rev. Fluid Mech.*, Vol. 33, pp445-490.

[4] Antoulas, A.C. (2005), Approximation of large-scale dynamical systems, Advances in Design and Control.

[5] Juang, J.-N. and Pappa, R.S. (1985) An eigensystem realization algorithm for modal parameter identification and model reduction. *Journal of Guidance, Control and Dynamics*, Vol. 8, No 5, pp620-627.

[6] Ma, Z., Ahuja, S. and Rowley, C. (2011)Reduced order models for control of fluids using the Eigensystem Realization Algorithm, Theor.

Comput. Fluid Dyn., Vol. 25, pp233-247.

[7] Gaitonde, A.L. and Jones, D.P. (2003) Reduced order state-space models from the pulse responses of a linearised CFD scheme, Int J. Numer. Meth. Fluids, **42**:581-606.

[8] Silva, W. and Raveh, D. (2001) Development of unsteady aerodynamic state-space models from CFD-based pulse responses, 19th AIAA Applied Aerodynamics Conference, Fluid Dynamics and Co-located Conferences, http://dx.doi.org/10.2514/6.2001-1213

[9] Silva, W. and Bartels, R.E. (2004) Development of reduced-order models for aeroelastic analysis and flutter prediction using the CFL3Dv6.0 code, Journal of Fluids and Structures Vol. 19, No 6, pp729âŁ“745. doi:10.1016/j.jfluidstructs.2004.03.004

[10] Wales, C.J., Gaitonde, A.L. and Jones, D.P. (2013) Stabilisation of Reduced Order Models via Restarting, IJNMF, Vol 73, Issue 6, pp578-599.

[11] Kung, S.-Y., Arun, K.S. and Bhaskar Rao, D.V. (1978) A new identification and model reduction algorithm via singular value decomposition, Proceedings of the 12th Asilomar conference on circuits, systems and computers, Pacific Grove.

[12] Gaitonde, A.L. and Jones, D.P. (2006) Study of linear response identification techniques and reduced order model generation for a 2D CFD scheme, Int. J. Numer. Meth. Fluids, Vol. 52, No 12, pp1361-1402.

[13] Gaitonde, A.L. and Jones,D.P. (2006) Study of techniques for obtaining continuous models from 2D discrete reduced-order state-space CFD models, Int. J. Numer. Meth. Fluids, Vol. 52, No 11, pp1247-1275.

[14] Friswell, M.I. and Mottershead, J.E. (1995) Finite Element Model Updating in Structural Dynamics, Kluwer Academic Publishers.

[15] Mottershead, J. E. and Friswell, M. I. (1993) Model updating in structural dynamics: a survey, Journal of Sound and Vibration, Vol 167, pp347-375.

[16] Mottershead, J. E., Link, M. and Friswell, M. I. (2011) The Sensitivity Method in Finite Element Model Updating: A Tutorial., Mechanical Systems and Signal Processing, Vol 25, No 7, pp 2275-2296.

[17] Friswell, M. I., Mottershead, J. E. and Ahmadian, H. (2001) Finite-Element Model Updating Using Experimental Test Data: Parametrization and Regularization, Phil. Trans. R. Soc., Vol 359, pp169-186.

[18] Rodden, W.P., Taylor, P.F. and McIntosh, S.C. (1998) Further Refinement of the Subsonic Doublet-Lattice Method, Journal of Aircraft Vol 35, No. 5, pp720-727.

[19] Palacios, R., Climent, H., Karlsson, A. and Winzell, B. (2001) Assessment of strategies for correcting linear unsteady aerodynamics using CFD or experimental results. International Forum on Aeroelasticity and Structural Dynamics (IFASD), 66(17 2), 2001

[20] Brink-Spalink, J. and Bruns, J.M. (2000) Correction of unsteady aerodynamic influence coefficients using experimental or CFD data, 41st AIAA Conference, AIAA-2000-1489

[21] Thormann, R. and Dimitrov, D. (2014) Correction of aerodynamic influence matrices for transonic flow, CEAS Aeronautical Journal, Vol 5, Issue 4, pp435-446.

[22] Wilcox K., (2006) Unsteady flow sensing and estimation via the gappy proper orthogonal decomposition, Computers and Fluids,Vol 35, No 2, pp208-226.

[23] Goetz S., Mifsud M., Aumann P. and Hansen L., (2010) Fusing experimental and computational data for harmonized aerodynamic data, CASE Newsletter.

[24] Jameson, A., Schmidt, W. and Turkel, E. (1981) Numerical solution of the Euler equations by finite volume method using Runge-Kutta time-stepping schemes, AIAA 14th Fluid and Plasma Dynamic Conference, AIAA-1981-1259.

[25] Kroll, N. and Jain, R.K. (1987) Solution of two-dimensional Euler equations - Experience with a finite volume code, Forschungsbericht, DFVLR-FB 87-41, Braunschweig, Germany.

[26] Gaitonde, A.L. (1994) A dual time method for the solution of the unsteady Euler equations. *Aeronautical Journal*; **98**(978).

[27] Jameson, A.J. (1991) Time dependent calculations using multigrid, with applications to unsteady flows past airfoils and wings. AIAA Paper 91-1596.

[28] Arnone, A., Liou, M.S. and Povinelli, L. (1995) Integration of Navier-Stokes equations using dual time stepping and a multigrid method, AIAA Journal, Vol 33.

[29] Griffiths L. (2014) Reduced order model updating, PhD Thesis, University of Bristol, http://laurencegriffiths.co.uk/research/.

[30] Cooley, J.W. and Tukey, J.W. (1965) An algorithm for the machine calculation of complex Fourier series, Mathematics of Computation, Vol 19, No 90, pp297-301.

[31] Moin, P. and Mahesh, K. (2007) Python for scientific computing, Compu. Sci. Eng., Vol 9, No 3, pp10-20.

[32] ERCOFTAC (2000)Quality and trust in industrial CFD - Best practice guidelines.

[33] Brereton, G.J., Reynolds, W.C. and Jayaraman, R. (2006) Response of a turbulent boundary layer to sinusoidal free-stream unsteadiness, Journal of Fluid Mechanics, Vol 221.

[34] Kersken, H. Frey, C., Voigt, C. and Ashcroft, G. (2012) Time-Linearized and Time-Accurate 3D RANS Methods for Aeroelastic Analysis in Turbomachinery, Journal of Turbomachinery, Vol. 134, No 5.

[35] Hall, K.C., Clark, W.S., Kersken, H., Frey, C., Voigt, C. and Ashcroft, G. (2000) A time-linearized Navier-Stokes analysis of stall flutter, Journal of Turbomachinery, Vol. 122, No 3, pp467-476.

[36] AGARD (1982) Compendium of unsteady measurements, AGARD-R-702, 1982.

[37] Landon, R.H. (1982) NACA 0012 oscillatory and transient pitching. Dataset 3 from AGARD-R-702.

[38] Wales C., Gaitonde A. and Jones D. (2016) Reduced-order modeling of gust responses, To be published in J. Aircraft, 2017.